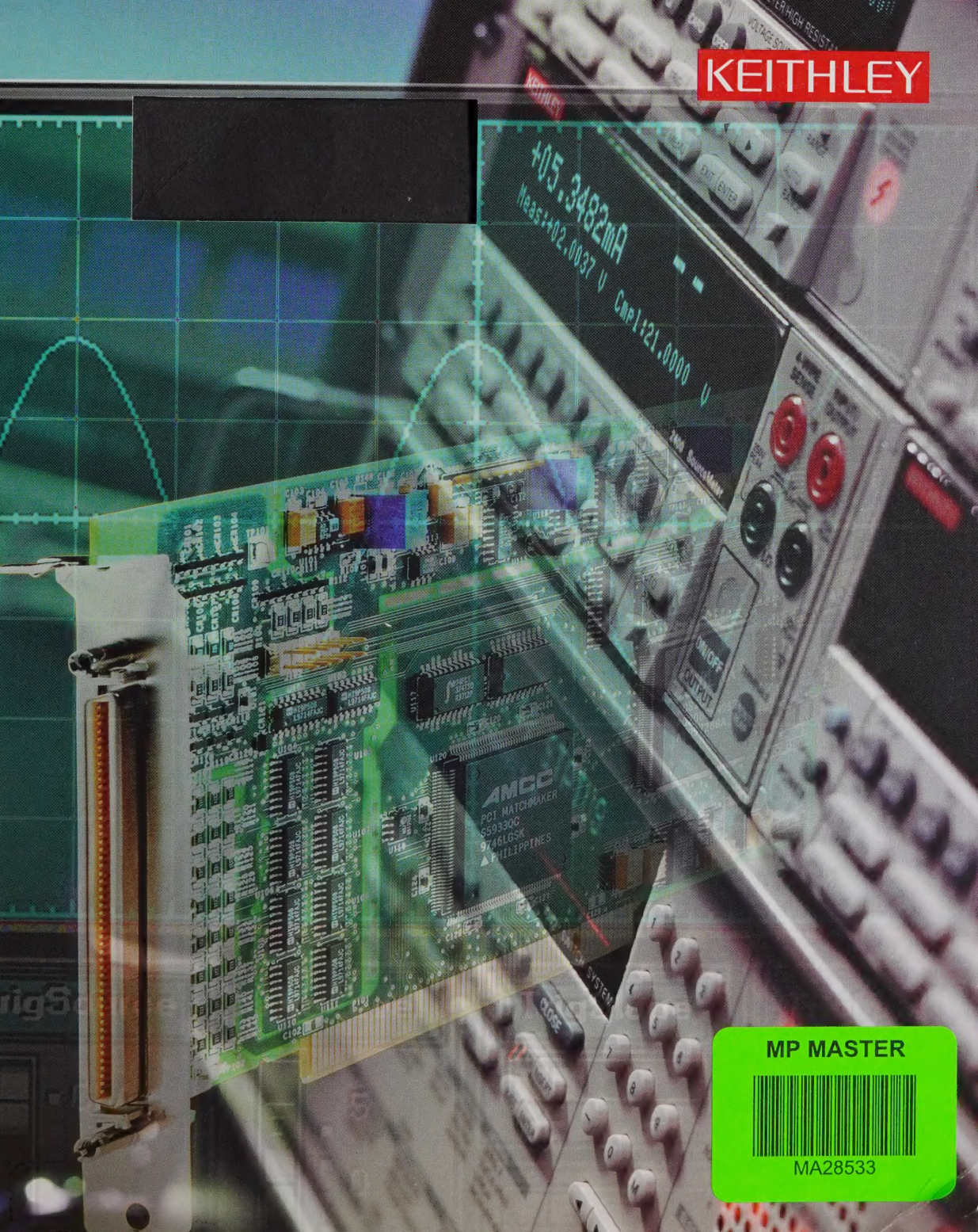

KEITHLEY
+05.3482mA
Meas:+02.0037 V
Comp:+21.0000 V
AMCC
PCI MATCHMAKER
S5933QC
9746LGSK
PHILIPPINES
MP MASTER
MA28533

Model 2510
TEC SourceMeter®
User's Manual

Contains Operating and Programming Information

KEITHLEY

WARRANTY

Keithley Instruments, Inc. warrants this product to be free from defects in material and workmanship for a period of 1 year from date of shipment.

Keithley Instruments, Inc. warrants the following items for 90 days from the date of shipment: probes, cables, rechargeable batteries, diskettes, and documentation.

During the warranty period, we will, at our option, either repair or replace any product that proves to be defective.

To exercise this warranty, write or call your local Keithley representative, or contact Keithley headquarters in Cleveland, Ohio. You will be given prompt assistance and return instructions. Send the product, transportation prepaid, to the indicated service facility. Repairs will be made and the product returned, transportation prepaid. Repaired or replaced products are warranted for the balance of the original warranty period, or at least 90 days.

LIMITATION OF WARRANTY

This warranty does not apply to defects resulting from product modification without Keithley's express written consent, or misuse of any product or part. This warranty also does not apply to fuses, software, non-rechargeable batteries, damage from battery leakage, or problems arising from normal wear or failure to follow instructions.

THIS WARRANTY IS IN LIEU OF ALL OTHER WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR USE. THE REMEDIES PROVIDED HEREIN ARE BUYER'S SOLE AND EXCLUSIVE REMEDIES.

NEITHER KEITHLEY INSTRUMENTS, INC. NOR ANY OF ITS EMPLOYEES SHALL BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF ITS INSTRUMENTS AND SOFTWARE EVEN IF KEITHLEY INSTRUMENTS, INC., HAS BEEN ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH DAMAGES. SUCH EXCLUDED DAMAGES SHALL INCLUDE, BUT ARE NOT LIMITED TO: COSTS OF REMOVAL AND INSTALLATION, LOSSES SUSTAINED AS THE RESULT OF INJURY TO ANY PERSON, OR DAMAGE TO PROPERTY.

KEITHLEY

Keithley Instruments, Inc. • 28775 Aurora Road • Cleveland, OH 44139 • 440-248-0400 • Fax: 440-248-6168 • http://www.keithley.com

BELGIUM:	Keithley Instruments B.V.	Bergensesteenweg 709 • B-1600 Sint-Pieters-Leeuw • 02/363 00 40 • Fax: 02/363 00 64
CHINA:	Keithley Instruments China	Yuan Chen Xin Building, Room 705 • 12 Yumin Road, Dewai, Madian • Beijing 100029 • 8610-62022886 • Fax: 8610-62022892
FRANCE:	Keithley Instruments Sarl	3, allée des Garays • 91127 Palaiseau Cedex • 01-64 53 20 20 • Fax: 01-60 11 77 26
GERMANY:	Keithley Instruments GmbH	Landsberger Strasse 65 • 82110 Germering • 089/84 93 07-40 • Fax: 089/84 93 07-34
GREAT BRITAIN:	Keithley Instruments Ltd	The Minster • 58 Portman Road • Reading, Berkshire RG30 1EA • 0118-9 57 56 66 • Fax: 0118-9 59 64 69
INDIA:	Keithley Instruments GmbH	Flat 2B, WILOCRISSA • 14, Rest House Crescent • Bangalore 560 001 • 91-80-509-1320/21 • Fax: 91-80-509-1322
ITALY:	Keithley Instruments s.r.l.	Viale S. Gimignano, 38 • 20146 Milano • 02-48 39 16 01 • Fax: 02-48 30 22 74
NETHERLANDS:	Keithley Instruments B.V.	Postbus 559 • 4200 AN Gorinchem • 0183-635333 • Fax: 0183-630821
SWITZERLAND:	Keithley Instruments SA	Kriesbachstrasse 4 • 8600 Dübendorf • 01-821 94 44 • Fax: 01-820 30 81
TAIWAN:	Keithley Instruments Taiwan	1 Fl. 85 Po Ai Street • Hsinchu, Taiwan, R.O.C. • 886-3572-9077• Fax: 886-3572-903

9/00

Model 2510 TEC SourceMeter®
User's Manual

©2000, Keithley Instruments, Inc.
All rights reserved.
Cleveland, Ohio, U.S.A.
Third Printing, April 2000
Document Number: 2510-900-01 Rev. B

Manual Print History

The print history shown below lists the printing dates of all Revisions and Addenda created for this manual. The Revision Level letter increases alphabetically as the manual undergoes subsequent updates. Addenda, which are released between Revisions, contain important change information that the user should incorporate immediately into the manual. Addenda are numbered sequentially. When a new Revision is created, all Addenda associated with the previous Revision of the manual are incorporated into the new Revision of the manual. Each new Revision includes a revised copy of this print history page.

Revision A (Document Number 2510-900-01) ..... January 2000
Revision A1 (Document Number 2510-900-01) ..... March 2000
Revision B (Document Number 2510-900-01) ..... April 2000

All Keithley product names are trademarks or registered trademarks of Keithley Instruments, Inc.
Other brand names are trademarks or registered trademarks of their respective holders.

Safety Precautions

The following safety precautions should be observed before using this product and any associated instrumentation. Although some instruments and accessories would normally be used with non-hazardous voltages, there are situations where hazardous conditions may be present.

This product is intended for use by qualified personnel who recognize shock hazards and are familiar with the safety precautions required to avoid possible injury. Read the operating information carefully before using the product.

The types of product users are:

Responsible body is the individual or group responsible for the use and maintenance of equipment, for ensuring that the equipment is operated within its specifications and operating limits, and for ensuring that operators are adequately trained.

Operators use the product for its intended function. They must be trained in electrical safety procedures and proper use of the instrument. They must be protected from electric shock and contact with hazardous live circuits.

Maintenance personnel perform routine procedures on the product to keep it operating, for example, setting the line voltage or replacing consumable materials. Maintenance procedures are described in the manual. The procedures explicitly state if the operator may perform them. Otherwise, they should be performed only by service personnel.

Service personnel are trained to work on live circuits, and perform safe installations and repairs of products. Only properly trained service personnel may perform installation and service procedures.

Exercise extreme caution when a shock hazard is present. Lethal voltage may be present on cable connector jacks or test fixtures. The American National Standards Institute (ANSI) states that a shock hazard exists when voltage levels greater than 30V RMS, 42.4V peak, or 60VDC are present. **A good safety practice is to expect that hazardous voltage is present in any unknown circuit before measuring.**

Users of this product must be protected from electric shock at all times. The responsible body must ensure that users are prevented access and/or insulated from every connection point. In some cases, connections must be exposed to potential human contact. Product users in these circumstances must be trained to protect themselves from the risk of electric shock. If the circuit is capable of operating at or above 1000 volts, **no conductive part of the circuit may be exposed.**

As described in the International Electrotechnical Commission (IEC) Standard IEC 664, digital multimeter measuring circuits (e.g., Keithley Models 175A, 199, 2000, 2001, 2002, and 2010) are Installation Category II. All other instruments' signal terminals are Installation Category I and must not be connected to mains.

Do not connect switching cards directly to unlimited power circuits. They are intended to be used with impedance limited sources. NEVER connect switching cards directly to AC mains. When connecting sources to switching cards, install protective devices to limit fault current and voltage to the card.

Before operating an instrument, make sure the line cord is connected to a properly grounded power receptacle. Inspect the connecting cables, test leads, and jumpers for possible wear, cracks, or breaks before each use.

For maximum safety, do not touch the product, test cables, or any other instruments while power is applied to the circuit under test. ALWAYS remove power from the entire test system and discharge any capacitors before: connecting or disconnecting cables or jumpers, installing or removing switching cards, or making internal changes, such as installing or removing jumpers.

Do not touch any object that could provide a current path to the common side of the circuit under test or power line (earth) ground. Always make measurements with dry hands while standing on a dry, insulated surface capable of withstanding the voltage being measured.

The instrument and accessories must be used in accordance with its specifications and operating instructions or the safety of the equipment may be impaired.

Do not exceed the maximum signal levels of the instruments and accessories, as defined in the specifications and operating information, and as shown on the instrument or test fixture panels, or switching card.

When fuses are used in a product, replace with same type and rating for continued protection against fire hazard.

Chassis connections must only be used as shield connections for measuring circuits, NOT as safety earth ground connections.

If you are using a test fixture, keep the lid closed while power is applied to the device under test. Safe operation requires the use of a lid interlock.

If a screw is present, connect it to safety earth ground using the wire recommended in the user documentation.

The symbol on an instrument indicates that the user should refer to the operating instructions located in the manual.

The symbol on an instrument shows that it can source or measure 1000 volts or more, including the combined effect of normal and common mode voltages. Use standard safety precautions to avoid personal contact with these voltages.

The **WARNING** heading in a manual explains dangers that might result in personal injury or death. Always read the associated information very carefully before performing the indicated procedure.

The **CAUTION** heading in a manual explains hazards that could damage the instrument. Such damage may invalidate the warranty.

Instrumentation and accessories shall not be connected to humans.

Before performing any maintenance, disconnect the line cord and all test cables.

To maintain protection from electric shock and fire, replacement components in mains circuits, including the power transformer, test leads, and input jacks, must be purchased from Keithley Instruments. Standard fuses, with applicable national safety approvals, may be used if the rating and type are the same. Other components that are not safety related may be purchased from other suppliers as long as they are equivalent to the original component. (Note that selected parts should be purchased only through Keithley Instruments to maintain accuracy and functionality of the product.) If you are unsure about the applicability of a replacement component, call a Keithley Instruments office for information.

To clean an instrument, use a damp cloth or mild, water based cleaner. Clean the exterior of the instrument only. Do not apply cleaner directly to the instrument or allow liquids to enter or spill on the instrument. Products that consist of a circuit board with no case or chassis (e.g., data acquisition board for installation into a computer) should never require cleaning if handled according to instructions. If the board becomes contaminated and operation is affected, the board should be returned to the factory for proper cleaning/servicing.

Table of Contents

1 Getting Started

General information 1-2
- Warranty information 1-2
- Contact information 1-2
- Manual addenda 1-2
- Safety symbols and terms 1-2
- Inspection 1-3
- Options and accessories 1-3

Product overview 1-5
Front and rear panel familiarization 1-6
- Front panel summary 1-6
- Rear panel summary 1-8

Power-up 1-9
- Line power connection 1-9
- Power-up sequence 1-9
- System identification 1-10
- Line frequency setting 1-10
- Fuse replacement 1-11

Display 1-11
- Display format 1-11
- Display readings 1-12
- Reading format 1-13
- Display examples 1-14
- Display messages 1-15
- ON/OFF indicator 1-16
- EDIT keys 1-16
- Status and error messages 1-16
- Front panel tests 1-16

Default settings 1-17
- Saving and restoring user setups 1-17
- Factory default settings 1-17

Menus 1-20
- Main menu 1-20
- Rules to navigate menus 1-22
- Configuration menus 1-22

2 Connections

Input/output connections 2-2
- Input/output connector 2-2
- 2-wire connections 2-2

Sense selection 2-4
- OUTPUT sensing 2-4
- INPUT sensing 2-4
- Sensing considerations 2-4

Ground connect mode 2-6
AC ohms measurement 2-6

3 Basic Operation

Safety precautions 3-2
Operation overview 3-2
- Control characteristics 3-2
- Measurement characteristics 3-2
- Limit characteristics 3-3

Configuring functions 3-3
- Configuring temperature 3-3
- Configuring voltage 3-5
- Configuring current 3-5
- Configuring resistance 3-6
- Configuring output 3-8
- Configuring setpoint tolerance 3-8

Adjusting setpoints 3-9
Basic front panel control-measure procedure 3-10
Basic remote control-measure procedure 3-11
Protection limits 3-13
- Temperature protection limits 3-14
- Voltage protection limit 3-14
- Current protection limit 3-14
- Resistance protection limits 3-14
- Voltage and current limit operating boundaries 3-15

Setpoint tolerance 3-15
- Ranges 3-16
- Operation 3-16
- Setpoint tolerance indicator 3-16
- Setpoint tolerance operation 3-17

Temperature sensors 3-18
- Open and shorted lead status indications 3-18
- Solid-state sensor restrictions 3-18
- Thermistor sensor ranges 3-18

4 PID Control Concepts

Temperature control model 4-2
Temperature control methods 4-2
- On-off control 4-3
- Proportional (P) control 4-3
- Proportional-derivative (PD) control 4-4
- Proportional-integral-derivative (PID) control 4-4

PID tuning 4-5
Sensor coefficients 4-6
- Thermistor coefficients 4-6
- RTD coefficients 4-6

5 Digital I/O Port and Output Enable

Digital I/O port 5-2
- Port configuration 5-2
- Digital output configuration 5-3
- Controlling digital output lines 5-4

Output enable line 5-6
- Overview 5-6
- Activating output enable 5-6

6 Remote Operations

Differences: remote vs. local operation 6-2
- Local-to-remote transition 6-2
- Remote-to-local transition 6-2
- Setting interface parameters 6-2

Selecting an interface 6-2
GPIB operation 6-3
- GPIB standards 6-3
- GPIB connections 6-4
- Primary address 6-5

RS-232 interface operation 6-6
- Sending and receiving data 6-6
- Baud rate 6-6
- Data bits and parity 6-6
- Terminator 6-7
- Flow control (signal handshaking) 6-7
- RS-232 connections 6-8

Front panel GPIB operation 6-9
- Error and status messages 6-9
- GPIB status indicators 6-9
- DISPLAY TOGGLE/LOCAL key 6-10

General bus commands 6-11
REN (remote enable) 6-11
IFC (interface clear) 6-11
LLO (local lockout) 6-11
GTL (go to local) 6-12
DCL (device clear) 6-12
SDC (selective device clear) 6-12
GET (group execute trigger) 6-12
SPE, SPD (serial polling) 6-12
Programming syntax 6-12
Command words 6-12
Query commands 6-14
Case sensitivity 6-14
Leading colon 6-15
Long-form and short-form versions 6-15
Short-form rules 6-15
Program messages 6-16
Response messages 6-18
Message exchange protocol 6-18

7 Status Structure

Overview 7-2
Status byte and SRQ 7-2
Status register sets 7-2
Queues 7-2
Clearing registers and queues 7-4
Programming and reading registers 7-5
Programming enable registers 7-5
Reading registers 7-6
Status byte and service request (SRQ) 7-7
Status byte register 7-8
Service request enable register 7-9
Serial polling and SRQ 7-9
SPE, SPD (serial polling) 7-9
Status byte and service request commands 7-10
Status register sets 7-11
Register bit descriptions 7-11
Condition registers 7-17
Event registers 7-17
Event enable registers 7-18
Queues 7-19
Output queue 7-19
Error queue 7-20

8 Common Commands

Command summary 8-2
Command reference 8-3
*IDN? — identification query 8-3
*OPC — operation complete 8-3
*OPC? — operation complete query 8-3
*OPT? — option query 8-3
*SAV <NRf> — save 8-4
*RCL <NRf> — recall 8-4
*RST — reset 8-4
*TRG — trigger 8-4
*TST? — self-test query 8-4
*WAI — wait-to-continue 8-5

9 SCPI Signal-Oriented Measurement Commands

Command summary 9-2
Acquiring readings 9-2
:FETCh? 9-2
:MEASure[:<function>]? 9-3
:READ? 9-4
:INITiate[:IMMediate] 9-4
Command examples 9-4

10 SCPI Command Reference

Reference tables 10-2
DISPlay subsystem 10-14
Control display 10-14
Read display 10-14
Define :TEXT messages 10-15
ASCII display values 10-16
FORMat subsystem 10-18
Data format 10-18
Data elements 10-20
Byte order 10-22
Status register format 10-23
OUTPut subsystem 10-23
Turn source on or off 10-23
Output enable line control 10-24
SENSe1 subsystem 10-24
Current function 10-24
Resistance function 10-25
Temperature function 10-26

SOURce[1] subsystem 10-31
Control source output-off 10-31
Select source function 10-31
Current function 10-32
Resistance function 10-33
Temperature function 10-35
Voltage function 10-38
Setpoint tolerance 10-39
SOURce2 subsystem 10-40
STATus subsystem 10-41
Read event registers 10-41
Program event enable registers 10-42
Read condition registers 10-42
Select default conditions 10-42
Error queue 10-43
SYSTem subsystem 10-44
Default conditions 10-44
Select power line frequency setting 10-44
Error queue 10-45
Simulate key presses 10-46
Read version of SCPI standard 10-47
Reset timestamp 10-47
2-wire/4-wire sense mode 10-47
Ground connect mode 10-48
RS-232 interface 10-48
Trigger subsystem 10-49
Initiate control/measure cycle 10-49
Abort source/measure cycle 10-49
UNIT subsystem 10-49

A **Specifications**

B **Status and Error Messages**

Introduction B-2
Status and error messages B-2
Eliminating common SCPI errors B-7

C **IEEE-488 Bus Overview**

Introduction C-2
Bus description C-2
Bus lines C-4
Data lines C-4
Bus management lines C-4
Handshake lines C-5

Bus commands C-6
Uniline commands C-7
Universal multiline commands C-7
Addressed multiline commands C-8
Address commands C-8
Unaddress commands C-8
Common commands C-9
SCPI commands C-9
Command codes C-9
Typical command sequences C-11
IEEE command groups C-12
Interface function codes C-13

D IEEE-488 and SCPI Conformance Information

Introduction D-2

E Example Programs

Introduction E-2
Program requirements E-2
Computer hardware requirements E-2
Software requirements E-2
General program instructions E-2
Basic temperature control program E-3
Instrument setup E-3
Requested readings E-3
Setpoint tolerance temperature sweep E-4
Overview E-4
Enabling SRQ on setpoint tolerance E-5
Temperature sweep program summary E-5
Program 1: Basic temperature control E-6
Program 2: Setpoint tolerance temperature sweep E-7

F GPIB 488.1 Protocol

Introduction F-2
Selecting the 488.1 protocol F-2
Protocol differences F-3
Message exchange protocol (MEP) F-3
Using SCPI-based programs F-3
Bus hold-off F-4
Trigger-on-talk F-4
Message available F-4
General operation notes F-4

List of Illustrations

1 Getting Started

Figure 1-1 Model 2510 front panel 1-6
Figure 1-2 Model 2510 rear panel 1-8
Figure 1-3 Main menu tree 1-23

2 Connections

Figure 2-1 2-wire input/output connections 2-2
Figure 2-2 4-wire input/output connections 2-3
Figure 2-3 Sensing methods 2-5
Figure 2-4 Ground connect mode 2-6
Figure 2-5 AC ohms measurement 2-6

3 Basic Operation

Figure 3-1 Voltage and current limit operating boundaries 3-15
Figure 3-2 Setpoint tolerance operation 3-17

4 PID Control Concepts

Figure 4-1 Temperature control model 4-2
Figure 4-2 Proportional control characteristics 4-3
Figure 4-3 PID control characteristics 4-5

5 Digital I/O Port and Output Enable

Figure 5-1 Digital I/O port 5-2
Figure 5-2 Sink operation 5-3
Figure 5-3 Source operation 5-4
Figure 5-4 Using output enable 5-7

6 Remote Operations

Figure 6-1 IEEE-488 connector 6-4
Figure 6-2 IEEE-488 connections 6-4
Figure 6-3 IEEE-488 and RS-232 connector locations 6-5
Figure 6-4 RS-232 interface connector 6-8

7 Status Structure

Figure 7-1 Model 2510 status 7-3
Figure 7-2 16-bit status register 7-5
Figure 7-3 Status byte and service request (SRQ) 7-7
Figure 7-4 Standard event status 7-12
Figure 7-5 Operation event status 7-13
Figure 7-6 Measurement event status 7-15
Figure 7-7 Questionable event status 7-16

10 SCPI Command Reference

Figure 10-1 ASCII data format 10-18
Figure 10-2 IEEE-754 single precision data format (32 data bits) 10-19

C IEEE-488 Bus Overview

Figure C-1 IEEE-488 bus configuration C-3
Figure C-2 IEEE-488 handshake sequence C-5
Figure C-3 Command codes C-10

E Example Programs

Figure E-1 Setpoint tolerance temperature sweep E-4

List of Tables

1 Getting Started

Table 1-1 Display examples 1-14
Table 1-2 Display messages 1-15
Table 1-3 Factory front panel default settings 1-18
Table 1-4 Main menu 1-20
Table 1-5 Temperature configuration menu 1-24
Table 1-6 Voltage source configuration menu 1-24
Table 1-7 Current source configuration menu 1-25
Table 1-8 Resistance configuration menu 1-25
Table 1-9 Output configuration menu 1-26
Table 1-10 Setpoint tolerance configuration menu 1-26

3 Basic Operation

Table 3-1 Temperature configuration menu 3-4
Table 3-2 Voltage source configuration menu 3-5
Table 3-3 Current source configuration menu 3-5
Table 3-4 Resistance configuration menu 3-7
Table 3-5 Output configuration menu 3-8
Table 3-6 Setpoint tolerance configuration menu 3-9
Table 3-7 Setpoint adjustment ranges 3-9
Table 3-8 Protection limit characteristics 3-13
Table 3-9 Setpoint tolerance ranges 3-16
Table 3-10 Sensor open lead and shorted lead ranges 3-18

5 Digital I/O Port and Output Enable

Table 5-1 Digital output line settings 5-5

6 Remote Operations

Table 6-1 RS-232 connector pinout 6-8
Table 6-2 PC serial port pinout 6-9
Table 6-3 General bus commands 6-11

7 Status Structure

Table 7-1 Common and SCPI commands to reset registers and clear queues 7-4
Table 7-2 Data format commands for reading status registers 7-6
Table 7-3 Status byte and service request enable register commands 7-10
Table 7-4 Status byte programming example 7-10
Table 7-5 Condition register commands 7-17

Table 7-6 Event register commands 7-17
Table 7-7 Event enable registers commands 7-18
Table 7-8 Program and read register programming example 7-19
Table 7-9 Error queue commands 7-21

8 Common Commands

Table 8-1 IEEE-488.2 common commands and queries 8-2

9 SCPI Signal-Oriented Measurement Commands

Table 9-1 Signal-oriented measurement command summary 9-2

10 SCPI Command Reference

Table 10-1 :DISPlay subsystem commands 10-3
Table 10-2 :FORMat subsystem commands 10-4
Table 10-3 OUTPut subsystem commands 10-4
Table 10-4 :SENSe[1] subsystem commands 10-5
Table 10-5 :SOURce[1] subsystem commands 10-7
Table 10-6 :SOURce2 subsystem commands 10-11
Table 10-7 :STATus subsystem commands 10-11
Table 10-8 :SYSTem subsystem commands 10-12
Table 10-9 Trigger subsystem commands 10-13
Table 10-10 :UNIT subsystem commands 10-13

B Status and Error Messages

Table B-1 Status and error messages B-3

C IEEE-488 Bus Overview

Table C-1 IEEE-488 bus command summary C-6
Table C-2 Hexadecimal and decimal command codes C-9
Table C-3 Typical addressed multiline command sequence C-11
Table C-4 Typical addressed common command sequence C-11
Table C-5 IEEE command groups C-12
Table C-6 Model 2510 interface function codes C-13

D IEEE-488 and SCPI Conformance Information

Table D-1 IEEE-488 documentation requirements D-3
Table D-2 Coupled commands D-4

1
Getting Started

- **General information** — Covers general information that includes warranty information, contact information, safety symbols and terms, inspection, and available options and accessories.
- **Product overview** — Summarizes the features and basic operating characteristics of the Model 2510.
- **Front and rear panel familiarization** — Summarizes the controls and connectors of the instrument.
- **Power-up** — Covers line power connection, line voltage settings, fuse replacement, and the power-up sequence.
- **Display** — Provides information about the Model 2510 display.
- **Default settings** — Covers factory default setups and saving and recalling user setups.
- **Menus** — Covers the main and configuration menus as well as rules to navigate menus.

General information

Warranty information

Warranty information is located at the front of this manual. Should your Model 2510 require warranty service, contact the Keithley representative or authorized repair facility in your area for further information. When returning the instrument for repair, be sure to fill out and include the service form at the back of this manual to provide the repair facility with the necessary information.

Contact information

Worldwide phone numbers are listed at the front of this manual. If you have any questions, please contact your local Keithley representative or call one of our Application Engineers at 1-800-348-3735 (U.S. and Canada only).

Manual addenda

Any improvements or changes concerning the instrument or manual will be explained in an addendum included with the manual. Be sure to note these changes and incorporate them into the manual.

Safety symbols and terms

The following symbols and terms may be found on the instrument or used in this manual.

The symbol on an instrument indicates that the user should refer to the operating instructions located in the manual.

The symbol on the instrument shows that high voltage may be present on the terminal(s). Use standard safety precautions to avoid personal contact with these voltages.

The **WARNING** heading used in this manual explains dangers that might result in personal injury or death. Always read the associated information very carefully before performing the indicated procedure.

The **CAUTION** heading used in this manual explains hazards that could damage the instrument. Such damage may invalidate the warranty.

Inspection

The Model 2510 was carefully inspected electrically and mechanically before shipment. After unpacking all items from the shipping carton, check for any obvious signs of physical damage that may have occurred during transit. (There may be a protective film over the display lens, which can be removed.) Report any damage to the shipping agent immediately. Save the original packing carton for possible future shipment. The following items are included with every Model 2510 order:

- Model 2510 with line cord
- Mating input/output connector (Keithley part no. CS-846)
- Accessories as ordered
- Certificate of calibration
- User's Manual

If an additional manual is required, order the appropriate manual package (for example, 2510-900-00). The manual packages include a manual and any pertinent addenda.

Options and accessories

The following options and accessories are available from Keithley for use with the Model 2510.

INPUT/OUTPUT mating connector

One mating connector for the rear panel INPUT/OUTPUT connector is supplied. Additional part number CS-846 connectors can be ordered from Keithley.

Cables and adapters

Models 7007-1 and 7007-2 shielded GPIB cables — Connect the Model 2510 to the GPIB bus using shielded cables and connectors to reduce electromagnetic interference (EMI). The Model 7007-1 is 1m long; the Model 7007-2 is 2m long.

Model 7009-5 shielded RS-232 cable — Connects the Model 2510 to a computer serial port using shielded cable and connectors to reduce EMI.

Rack mount kits

Model 4288-1 single fixed rack mount kit — Mounts a single Model 2510 in a standard 19-inch rack.

Model 4288-2 side-by-side rack mount kit — Mounts two instruments (Models 182, 428, 486, 487, 2000, 2001, 2002, 2010, 2015, 2016, 2400, 2410, 2420, 2430, 2510, 6430, 6517, 7001) side-by-side in a standard 19-inch rack.

Model 4288-3 side-by-side rack mount kit — Mounts a Model 2510 and a Model 199 side-by-side in a standard 19-inch rack.

Model 4288-4 side-by-side rack mount kit — Mounts a Model 2510 and a 5.25-inch instrument (Models 195A, 196, 220, 224, 230, 263, 595, 614, 617, 705, 740, 775, etc.) side-by-side in a standard 19-inch rack.

Model 4288-5 dual fixed rack mounting kit — Mounts a Model 2510 and another 3½-inch high instrument (Model 182, 428, 486, 487, 2000, 2010, 2400, 2410, 2420, 2430, 6430, or 7001), side-by-side in a standard 19-inch rack.

Carrying case

Model 1050 padded carrying case — A carrying case for a Model 2510. Includes handles and a shoulder strap.

Product overview

The Model 2510 TEC SourceMeter has the following operating characteristics:

- Thermoelectric cooler source range: ±10V DC at up to ±5A DC.
- Temperature, voltage, current, and resistance control functions.
- Software-controlled PID loop.
- Several measurement functions including operating resistance, voltage, current, and power, as well as AC resistance.
- Compatible with RTD, thermistor, and solid-state thermal feedback elements.
- Built-in IEEE-488 and RS-232 interfaces for remote operation.
- Output enable circuit to automatically remove source signal when a test fixture lid is open.
- Digital I/O port allows control of other instruments.
- Closed-cover calibration — The instrument can be calibrated either from the front panel or remote interface.

Front and rear panel familiarization

Front panel summary

The front panel of the Model 2510 is shown in Figure 1-1. The following abbreviated information should be reviewed before operating the instrument. See Section 3 for more detailed information.

Figure 1-1
Model 2510 front panel

Function keys:

T	Temperature function.
V	Voltage function.
I	Current function.
R	Resistance function.

EDIT keys:

▲ and ▼	Increase/decrease displayed value, scroll through menu.
◀ and ▶	Move display cursor left or right.

Operation keys:

DISPLAY TOGGLE/LOCAL	Toggle displayed values, cancel remote operation, restore front panel controls.
CONFIG	Press CONFIG and then T, V, I, R, EDIT or OUTPUT key to configure function.
MENU	Access and configure Main Menu selections.
EXIT	Cancel selection or back out of menu structures.
ENTER	Accept selection or value.

Annunciators:

EDIT	Instrument in edit mode.
ERR	Questionable reading, invalid cal step.
REM	Instrument in GPIB remote mode.
TALK	Instrument addressed to talk over GPIB.
LSTN	Instrument addressed to listen over GPIB.
SRQ	Service request over GPIB.
4W	Indicates 4-wire sense mode enabled.
ARM	Operations being performed.
REAR	Ground connect mode enabled.
*	Setpoint tolerance reached.

Power controls:

OUTPUT	Turns the control source OUTPUT on or off. Integrated red/green/yellow indicator shows heating/cooling status.
POWER	Turns main unit power on or off.

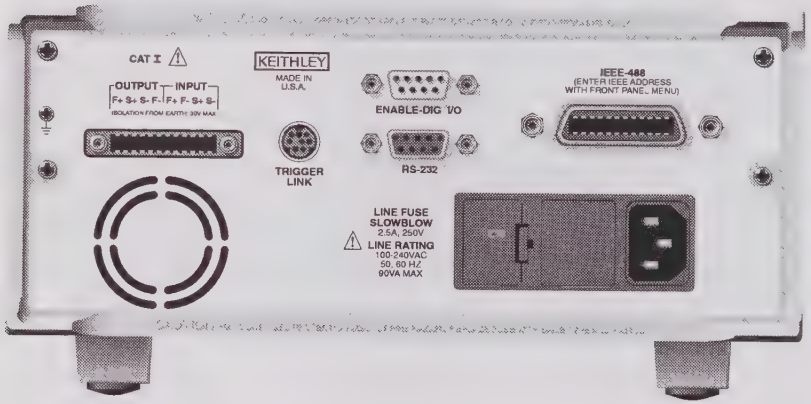
Handle:

Pull out and rotate to desired position.

Rear panel summary

The rear panel of the Model 2510 is shown in Figure 1-2. The following abbreviated information should be reviewed before operating the instrument.

Figure 1-2
Model 2510 rear panel



Input/output connector:

OUTPUT terminals	Source output connections to thermoelectric cooler.
INPUT terminals	Temperature sensor input connections.

Digital I/O port:

ENABLE — DIG I/O	Male DB-9 connector for digital output lines and output enable.

Power module:

Contains the AC line receptacle and the power line fuse.

Trigger link connector:

TRIGGER LINK	8-pin micro-DIN connector for sending and receiving trigger pulses. (Not currently implemented).

RS-232 connector:

RS-232	Connector for RS-232 remote operation. Use a straight through (not null modem) DB-9 cable such as a Keithley Model 7009-5.

GPIB connector:

IEEE-488 INTERFACE	Connector for GPIB remote operation. Use a shielded cable (Model 7007-1 or 7007-2).

Ground screw:

⏚ Allows easy access to chassis ground.

Fan:

Internal fan turns on when the internal temperature reaches 50°C. Keep ventilation slots free of obstructions to avoid overheating.

Power-up

Line power connection

The Model 2510 operates from a line voltage in the range of 100 to 240V at a frequency of 50 to 60Hz. Line voltage and line frequency are automatically sensed. Therefore, there are no switches to set. Before connecting the unit to line power, check to be sure the operating voltage in your area is compatible.

CAUTION **Operating the instrument on an incorrect line voltage may cause damage, possibly voiding the warranty.**

Perform the following steps to connect the Model 2510 to line power and turn it on:

1. Before plugging in the power cord, make sure the front panel POWER switch is in the off (0) position.
2. Connect the female end of the supplied power cord to the AC receptacle on the rear panel.
3. Connect the other end of the supplied power cord to a properly grounded AC outlet.

WARNING **The power cord supplied with the Model 2510 contains a separate ground for use with grounded outlets. When proper connections are made, instrument chassis is connected to power line ground through the ground wire in the power cord. Failure to use a grounded outlet may result in personal injury or death due to electric shock.**

4. Turn on the instrument by pressing the front panel power switch to the on (1) position.

Power-up sequence

On power-up, the Model 2510 performs self-tests on its EPROM and RAM and momentarily lights all segments and annunciators. If a failure is detected, the instrument momentarily displays an error message, and the ERR annunciator turns on.

If the instrument passes the self-tests, the firmware revision levels are displayed. For example:

REV A01 A02

where: A01 is the main board ROM revision.
A02 is the display board ROM revision.

Also displayed is the line frequency (50 or 60Hz). (If the wrong frequency is displayed, it can be set manually as covered below). The communication interface status is briefly displayed. If the IEEE-488 bus is the presently selected interface, the identification message will include the primary address. For example, if the primary address is 15 (factory default), the "IEEE Addr=15" message is displayed. (The Model 2510 does not support a secondary address.) If the RS-232 interface is selected, the "RS-232" message is displayed.

After the power-up sequence, the instrument goes to its normal display state with the output off (ON/OFF OUTPUT indicator light off). With the output off, the "OFF" message is displayed.

System identification

To obtain the serial number and revision information, use the MENU/GENERAL/SERIAL # selection. The top line displays the serial number; the bottom line displays the firmware revision and date of calibration. You can then press the ▶ key to display SCPI revision level and main board revision levels. You can also use *IDN? via remote (see Section 8).

Line frequency setting

The power line frequency and line frequency setting should be the same, or readings may be noisy. You can manually set the line frequency from the front panel as follows.

Set the line frequency from the front panel as follows:

1. Press the MENU key to display MAIN MENU.
2. Using the right arrow key, select AD-CTRL then press ENTER to display A/D CONTROLS.
3. Note that LINE-FREQUENCY is displayed, then press ENTER to display the LINE FREQUENCY menu.
4. Place the cursor on 50Hz or 60Hz, and press ENTER.
5. Press EXIT to return to normal display.

Fuse replacement

A rear panel fuse protects the power line input of the Model 2510. If the line fuse needs to be replaced, perform the following steps:

WARNING **Disconnect the line cord and all test leads and cables from the instrument before replacing the line fuse.**

CAUTION **For continued protection against fire or instrument damage, replace the fuse only with the type and rating listed. If the instrument repeatedly blows fuses, locate and correct the cause of the problem before replacing the fuse.**

1. The fuse is located in a drawer adjacent to the AC receptacle (Figure 1-2). At the bottom of the fuse drawer is a small tab. At this location, use a small bladed screwdriver to pry the fuse drawer open.
2. Slide the fuse drawer out to gain access to the fuse. Note that the fuse drawer does not pull all the way out of the input module.
3. Snap the fuse out of the drawer and replace it with the same type: 250V, 2.5A slow blow, 5 × 20mm, Keithley part number FU-106-2.5.
4. Push the fuse drawer back into the input module.

Display

Display format

The Model 2510 display is used primarily to program setpoint values and display measured readings. Annunciators, which are located along the top of the reading/message display, indicate various states of operation, as covered previously in "Front panel summary."

On power-up, the top (primary) display shows the function being controlled by the PID loop when the output is on (with the output off, "OFF" is displayed). The bottom display is used for setpoint values (on the left), and alternate reading information for the selected function (on the right) that can be selected with the DISPLAY TOGGLE/LOCAL key.

Display readings

Display readings depend on the selected function as discussed below.

Temperature function readings

The temperature function displays measured and setpoint temperatures. You can use the DISPLAY TOGGLE key to cycle among:

- Peltier (PEL) voltage
- Peltier current
- Peltier power
- Peltier resistance
- Sensor (RT) resistance

NOTE *The terms "Peltier" and "thermoelectric cooler" (TEC) are used interchangeably throughout this manual.*

Voltage function readings

The voltage function displays measured and setpoint TEC voltages. DISPLAY TOGGLE key cycles among:

- Temperature (T)
- Peltier current
- Peltier power
- Peltier resistance
- Sensor resistance

Current function readings

The current function displays measured and setpoint TEC currents, and you can use the DISPLAY TOGGLE key to cycle among:

- Temperature
- Peltier voltage
- Peltier power
- Peltier resistance
- Sensor resistance

Resistance function readings

The resistance function displays measured RTD and thermistor sensor resistances, as well as setpoint resistances, and you can use the DISPLAY TOGGLE key to cycle among:

- Temperature
- Peltier voltage
- Peltier current
- Peltier power
- Peltier resistance

Reading format

Reading information on the top line of the front panel display can be displayed using either engineering units or scientific notation in either fixed- or floating-point format. Use the GENERAL/NUMBERS selection of the main MENU to select the display format, as discussed under "Menus later" in this section.

Engineering units example: 1.23456µA

Scientific notation example: 1.23456e -6

See "Format subsystem" in Section 10 for remote reading formats.

Display examples

Typical display examples are shown in Table 1-1.

Table 1-1
Display examples

Function	Display toggle sequence*
Temperature	+025.000°C Setpoint: +025.000°C PEL:-00.072V +025.000°C Setpoint: +025.000°C PEL:-0.0030A +025.000°C Setpoint: +025.000°C PEL:+00.002W +025.000°C Setpoint: +025.000°C PEL:+002.40Ω +025.000°C Setpoint: +025.000°C RT:+000.04kΩ
Voltage	+01.000V Setpoint: +01.000V T:+030.630°C +01.000V Setpoint: +01.000V PEL:+0.3689A +01.000V Setpoint: +01.000V PEL:+00.368W +01.000V Setpoint: +01.000V PEL:+002.71Ω +01.000V Setpoint: +01.000V RT:+000.04kΩ
Current	+00.500A Setpoint: +00.500A T:+033.200°C +00.500A Setpoint: +00.500A PEL:+01.377V +00.500A Setpoint: +00.500A PEL:+00.689W +00.500A Setpoint: +00.500A PEL:+002.75Ω +00.500A Setpoint: +00.500A RT:+002.75kΩ

Table 1-1 (cont.)
Display examples

Function	Display toggle sequence*
Resistance	+010.00kΩ Setpoint: +010.000kΩ T:+033.200°C +010.00kΩ Setpoint: +010.000kΩ PEL:+01.377V +010.00kΩ Setpoint: +010.000kΩ PEL:+0.3689A +010.00kΩ Setpoint: +010.000kΩ PEL:+00.689W +010.00kΩ Setpoint: +010.000kΩ PEL:+002.75Ω

*Use DISPLAY TOGGLE key to cycle through displays for each function.

Display messages

Table 1-2 shows front panel messages that may be displayed during normal operation.

Table 1-2
Display messages

Display message	Description
OFF	OUTPUT turned off.
UNDER-TEMP	Temperature below lower limit.
OVER-TEMP	Temperature above upper limit.
ILIM	Current limit exceeded.
VLIM	Voltage limit exceeded.
UNDER-OHMS	Resistance below lower limit.
OVER-OHMS	Resistance above upper limit.
URR	Under resistance range (sensor).
ORR	Over resistance range (sensor).
OpenLead	Sensor lead open.
ShrtLead	Sensor lead shorted.

ON/OFF indicator

The ON/OFF indicator shows when the source output is turned on. This indicator will glow with different colors depending on the following conditions:

- Red = heating (+Peltier voltage)
- Green = cooling (-Peltier voltage)
- Red and green both on (yellow) = voltage overflow (not voltage limit).

EDIT keys

The EDIT keys allow you to program setpoint and other values, and scroll through menu selections as follows:

- When programming setpoint or other values, use the ▲ and ▼ keys to increment and decrement values; use the left and right arrow keys to select the cursor position. Pressing any one of these keys enters the setpoint editing mode for the selected function.
- When scrolling through menu selections, use the ◀ and ▶ keys to choose specific items.

See "Menus" later in this section for more details.

Status and error messages

During Model 2510 operation and programming, you will encounter a number of front panel messages. Status and error messages are displayed momentarily. See Appendix B for a list of status and error messages and ways to fix common errors.

Front panel tests

Use the TEST/DISPLAY-TESTS selection of the main MENU to test various aspects of the front panel. Test selections include:

- KEYS – Front panel keys are tested. Pressing a key displays a message that identifies that key. Pressing EXIT twice cancels this test.
- DISPLAY PATTERNS – Use this selection to turn on all display pixels and annunciators. Subsequent keypresses cycle through tests that turn off annunciators and corner pixels of each digit, turn on the rows of the top-left display digit, and turn on all annunciators and pixels of each digit in a sequential manner. Press EXIT to cancel this test.
- CHAR SET – This test displays special characters. Press EXIT to cancel the test.

See "Menus" later in this section for more menu information.

Default settings

By using appropriate menu selections, you can save and recall various instrument setups, define the power-on configuration, or restore factory defaults as outlined below.

Saving and restoring user setups

You can save and restore five of your own user setups as covered below. This feature provides a convenient way to save specific instrument configurations and then recall them as needed. Note that you can also set up the Model 2510 to restore a specific user setup at power-on (see "Power-on configuration" below). See Section 8 for information on using *SAV and *RCL to perform these operations by remote.

Saving setups

1. Select the various instrument operating modes you wish to save.
2. Press the MENU key, select SAVESETUP, then press ENTER.
3. Select SAVE, then press ENTER.
4. Select the setup position (0 to 4) to save, then press ENTER to complete the process.

Restoring setups

1. Press the MENU key, select SAVESETUP, then press ENTER.
2. Select RESTORE, then press ENTER.
3. Select the setup position (0 to 4) to restore, then press ENTER to complete the process.

Power-on configuration

You can also define which of the stored setups (factory default or user) the instrument assumes as the power-on configuration as follows:

1. Press the MENU key, select SAVESETUP, then press ENTER.
2. Select POWERON, then press ENTER.
3. From the SET POWER-ON DEFAULT menu, choose the power-on configuration: BENCH or GPIB (see below), or USER-SETUP-NUMBER.
4. If you chose to use a user setup as the power-on configuration, select the user setup number, then press ENTER.

Factory default settings

There are two sets of factory defaults, BENCH (front panel) and GPIB (remote). For front panel operation, BENCH and GPIB defaults are the same and are summarized in Table 1-3. You can restore these default conditions as follows:

1. Press the MENU key, select SAVESETUP, then press ENTER.
2. Select RESET, then press ENTER.
3. Select BENCH or GPIB defaults, then press ENTER to complete the process.

See the SCPI reference tables in Section 10 for remote default information.

Table 1-3
Factory front panel default settings

Setting	Front panel BENCH and GPIB default*
Control function	Temperature
Temperature function	
Setpoint	25°C (298.15K, 77°F)
Protection high limit	50°C (323.15K, 122°F)
Protection low limit	0°C (273.15K, 32°F)
Protection state	Enabled
Temperature units	Celsius
Sensor current	2.5mA (100Ω range, RTD) 833.3µA (1kΩ range) 100µA (10kΩ range) 33.3µA (100kΩ range)
Sensor auto current state	On
Sensor type	Thermistor
Sense mode	2-wire
RTD sensor parameters	
Type	PT385
Alpha (α)	0.00385
Beta (β)	0.111
Delta (δ)	1.507
Thermistor sensor parameters	
Range	10kΩ
A	1.12396e-3
B	2.30093e-4
C	7.06351e-8
Solid-state sensor parameters	
VSS gain	1e-2
VSS offset	0
ISS gain	1e-6
ISS offset	0
Proportional constant	20
Integral constant	0.6
Derivative constant	0
Voltage function	
Setpoint	0V
Voltage protection limit	10.5V
Proportional constant	10
Integral constant	20
Derivative constant	0

* BENCH defaults are the same as :SYSTem:PRESet defaults (Section 10). GPIB defaults are the same as *RST defaults (Section 8).
** Not affected by BENCH or GPIB RESET.

Table 1-3 (cont.)
Factory front panel default settings

Setting	Front panel BENCH and GPIB default*
Current function	
Setpoint	0A
Current protection limit	2A
Proportional constant	10
Integral constant	20
Derivative constant	0
Resistance function	
Setpoint	100Ω (RTD or 100Ω thermistor)
	1kΩ (1kΩ thermistor)
	10kΩ (10kΩ thermistor)
	100kΩ (100kΩ thermistor)
Protection high limit	200Ω (RTD or 100Ω thermistor)
	2kΩ (1kΩ thermistor)
	20kΩ (10kΩ thermistor)
	200kΩ (100kΩ thermistor)
Protection low limit	50Ω (RTD or 100Ω thermistor)
	500Ω (1kΩ thermistor)
	5kΩ (10kΩ thermistor),
	50kΩ (100kΩ thermistor)
Protection state	Enabled
Sensor current	2.5mA (100Ω range, RTD)
	833.3µA (1kΩ range)
	100µA (10kΩ range)
	33.3µA (100kΩ range)
Sensor current auto state	On
Sensor type	Thermistor
Sense mode	2-wire
Thermistor range	10kΩ
Proportional constant	20
Integral constant	0.6
Derivative constant	0
Output On/Off	Off
Digital I/O enable line	Off
Peltier (cooler) ground connect mode	Disabled
Digital I/O port output value	15
Display format**	Engineering units
Setpoint tolerance percentage	0.5%
Window size	5

* BENCH defaults are the same as :SYSTem:PRESet defaults (Section 10). GPIB defaults are the same as *RST defaults (Section 8).

** Not affected by BENCH or GPIB RESET.

Menus

The following paragraphs discuss the main menu, function configuration menus, and rules to navigate menus.

Main menu

Use the MENU key to access the Main Menu to select, configure, and/or perform various instrument operations, as shown in Table 1-4 and in Figure 1-3. Use the "Rules to navigate menus" discussed later to check and/or change Main Menu options.

Table 1-4
Main menu

Menu item[1]	Description	Parameters
SAVESETUP	Configure setup conditions.	
SAVE	Save present Model 2510 setup to memory location.	0 to 4
RESTORE	Return the Model 2510 to setup saved in memory.	0 to 4
POWERON	Select the power-on default setup.	
BENCH	Powers-on to BENCH defaults.	
GPIB	Powers-on to GPIB defaults.	
USER-SETUP-NUMBER	Powers-on to user setup.	0 to 4
RESET	Returns unit to BENCH or GPIB defaults.	
COMMUNICATION[2]	Select and configure remote interface.	
GPIB	Select GPIB (IEEE-488 Bus), set primary address. Select SCPI or 488.1 protocol (Appendix F).	0 to 30 (Default: 15)
RS-232	Select the RS-232 interface, set parameters.	57600, 38400, 19200, 9600, 4800, 2400, 1200, 600, 300
BAUD	Select baud rate.	
BITS	Select number of data bits.	7 or 8
PARITY	Select parity.	NONE, ODD, EVEN
TERMINATOR	Select terminator.	CR, CR+LF, LF, or LF+CR
FLOW-CTRL	Select flow control.	NONE or XON/XOFF

Table 1-4 (cont.)
Main menu

Menu item[1]	Description	Parameters
CAL[3]	Calibrate Model 2510. (See Service Manual).	
UNLOCK	Unlock calibration.	
EXECUTE	Execute calibration steps.	
VIEW DATES	Display calibration dates and count.	
SAVE	Save calibration constants.	
LOCK	Lock out calibration.	
CHANGE-PASSWORD	Change calibration password.	
TEST	Perform tests on Model 2510.	
DISPLAY TESTS[4]	Test front panel keys and display digits.	
KEYS	Test front panel keys.	
DISPLAY-PATTERNS	Test display pixels and annunciators.	
CHAR SET	Test display characters.	
A/D CTRL	Select line frequency.	50 or 60Hz
LINE-FREQUENCY	Set the line frequency.	
GENERAL	Select general operations.	
DIGOUT	Set Digital I/O port bit pattern.	0 to 15
SERIAL#	Display serial number, firmware revision, SCPI version.	
TIMESTAMP	Reset timestamp.	YES or NO
NUMBERS	Select engineering units or scientific notation display format (top display line only). Select FLOATING EXP or FIXED EXP for SCIENTIFIC.	ENGR, SCIENTIFIC

NOTES
[1]Top level menu choices indicated in bold. Indentation identifies each lower submenu level.
[2]When the remote operation interface selection (GPIB or RS-232) is changed, the Model 2510 performs a power-on reset. To check or change options of the selected interface, you must re-enter the menu structure.
[3]Only VIEW DATES is accessible when calibration is locked. Password required to unlock calibration. See Service Manual.
[4]Press EXIT key to cancel test.

Rules to navigate menus

Many functions and operations are configured from the front panel menus. Use the following rules to navigate through these configuration menus:

- A menu item is selected by placing the cursor on it and pressing ENTER. Cursor position is denoted by the blinking menu item or option. The EDIT left and right arrow keys control cursor position.
- A displayed arrow on the bottom line indicates there are one or more additional items (or messages) to select from. Use the appropriate EDIT cursor key to display them.
- A numeric value is changed by placing the cursor on the digit to be changed and using the EDIT ▲ and ▼ keys.
- A change is executed only when ENTER is pressed. Entering an invalid parameter generates an error, and the entry is ignored. However, entering an out-of-range value (too small or too large) selects the lower or upper limit, respectively.
- The EXIT key is used to back out of the menu structure. Any change that is not entered is cancelled when EXIT is pressed.

Configuration menus

There are five configuration menus that can be accessed by pressing the CONFIG key followed by the appropriate key (T, V, I, R, OUTPUT, or EDIT). Configuration menus on the following pages are available for these operating functions:

- Temperature (CONFIG T): Table 1-5.
- Voltage source (CONFIG V): Table 1-6.
- Current source (CONFIG I): Table 1-7.
- Resistance (CONFIG R): Table 1-8.
- Output (CONFIG OUTPUT): Table 1-9
- Setpoint tolerance (CONFIG EDIT): Table 1-10

NOTE *See "Configuring functions" in Section 3 for parameter information.*

Figure 1-3
Main menu tree

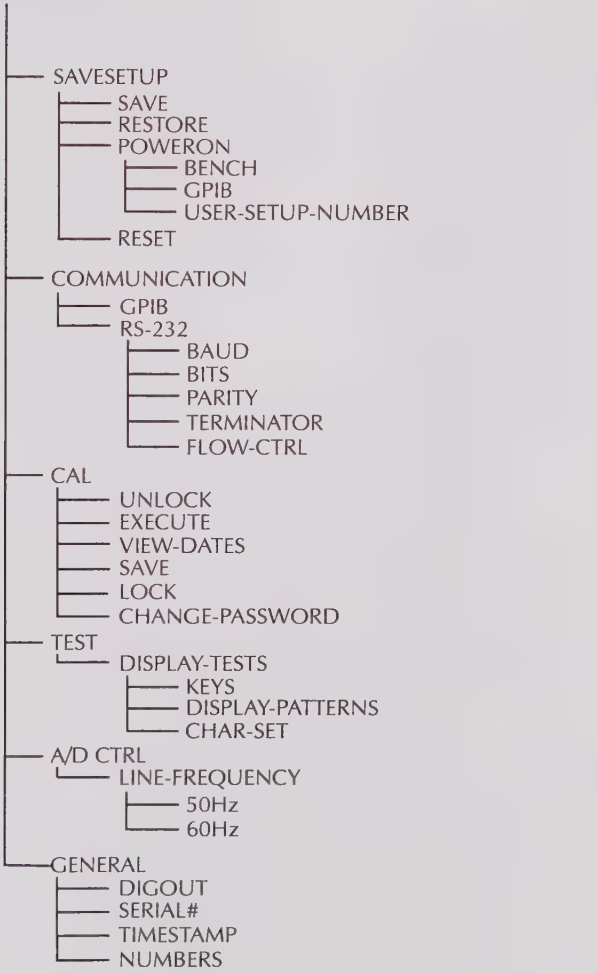


Table 1-5
Temperature configuration menu

Configuration menu item	Description
PROTECTION	Configure temperature protection.
CONTROL	ENABLE or DISABLE temperature protection.
LOLIM	Program lower temperature limit.
HILIM	Program higher temperature limit.
SENSOR-TYPE	Select sensor type and settings.
THERMISTOR	Program thermistor settings.
RANGE	Set thermistor range (100Ω, 1kΩ, 10kΩ, 100kΩ).
A	Program A coefficient.
B	Program B coefficient.
C	Program C coefficient.
I-SRC	Set thermistor current (AUTO, 3.3µA, 10µA, 33.3µA, 100µA, 833.3µA, 2.5mA)
SENSE-MODE	Select 2-WIRE or 4-WIRE sense mode.
RTD	Program RTD sensor parameters.
TYPE	Select PT385, PT3916, PT100, D100, F100, USER type. Program USER α, β, δ.
I-SRC	Set RTD current (AUTO, 3.3µA, 10µA, 33.3µA, 100µA, 833.3µA, 2.5mA).
SENSE-MODE	Select 2-WIRE or 4-WIRE sense mode.
I-SS	Program current solid-state sensor settings.
OFFSET	Program offset (K)
SCALE-FACTOR	Program slope (K/A).
V-SS	Program voltage solid-state sensor settings.
OFFSET	Program offset (K)
SCALE-FACTOR	Program slope (K/A).
UNITS	Select temperature units (°C, Kelvin, °F).
PID	Program PID loop constants for temperature control.
PROPORTIONAL	Set proportional constant.
INTEGRAL	Set integral constant.
DERIVATIVE	Set derivative constant.

Table 1-6
Voltage source configuration menu

Configuration menu item	Description
PROTECTION	Set voltage protection limit.
PID	Program PID loop constants for voltage control.
PROPORTIONAL	Set proportional constant.
INTEGRAL	Set integral constant.
DERIVATIVE	Set derivative constant.

Table 1-7
Current source configuration menu

Configuration menu item	Description
PROTECTION	Set current protection limit.
PID	Program PID loop constants for current control.
PROPORTIONAL	Set proportional constant.
INTEGRAL	Set integral constant.
DERIVATIVE	Set derivative constant.

Table 1-8
Resistance configuration menu

Configuration menu item	Description
PROTECTION	Configure resistance protection.
CONTROL	ENABLE or DISABLE resistance protection.
LOLIM	Program lower resistance limit.
HILIM	Program higher resistance limit.
SENSOR-TYPE	Select sensor type and settings.
THERMISTOR	Program thermistor settings.
RANGE	Set thermistor range (100Ω, 1kΩ, 10kΩ, 100kΩ).
I-SRC	Set thermistor current (AUTO, 3.3µA, 10µA, 33.3µA, 100µA, 833.3µA, 2.5mA).
SENSE-MODE	Select 2-WIRE or 4-WIRE sense mode.
RTD	Program RTD settings.
I-SRC	Set RTD current (AUTO, 3.3µA, 10µA, 33.3µA, 100µA, 833.3µA, 2.5mA).
SENSE-MODE	Select 2-WIRE or 4-WIRE sense mode.
PID	Program PID loop constants for resistance control.
PROPORTIONAL	Set proportional constant.
INTEGRAL	Set integral constant.
DERIVATIVE	Set derivative constant.

Table 1-9
Output configuration menu

Configuration menu item	Description
GND-CONNECT	Select input/output analog common (ground) connection.
DISABLE	Disable ground connection.
ENABLE	Enable ground connection.
ENABLE	Enable/disable Digital I/O port output enable line.
DISABLE	Inactivate output enable line.
ENABLE	Activate output enable line.

Table 1-10
Setpoint tolerance configuration menu

Configuration menu item	Description
SETPOINT-TOLERANCE	
PERCENTAGE	Percent of range for setpoint tolerance.
WINDOW-SIZE	Number readings to be within tolerance.

2
Connections

- **Input/output connections** — Discusses connections for the thermoelectric cooler and temperature sensor.
- **Sense selection** — Summarizes 2-wire and 4-wire sensing selection for both INPUT and OUTPUT terminals.
- **Ground connect mode** — Describes using the ground connect mode.

Input/output connections

Input/output connector

The input/output connector is located on the rear panel (see Figures 2-1 and 2-2). Terminals include:

- OUTPUT terminals: F+, F- (force), S+, and S- (sense) connections to thermoelectric cooler.
- INPUT terminals: F+, F- (force), S+, and S- (Kelvin sense) connections to temperature sensor.

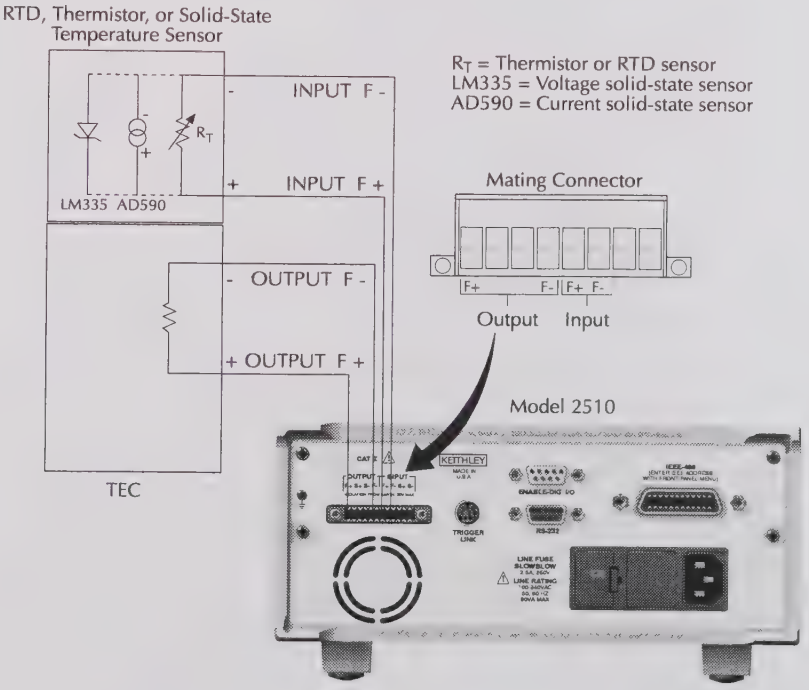
Use the supplied mating connector for connections (Keithley part number CS-846).

2-wire connections

Figure 2-1 shows basic 2-wire signal connections. Note that the OUTPUT F (force) terminals are connected to the thermoelectric cooler (TEC), while the INPUT F terminals are connected to the temperature sensor. 2-wire sensor connections are recommended for:

- Thermistor and RTD sensors with ≥1kΩ resistance.
- Both AD590 current and LM335 voltage type solid-state sensors.

Figure 2-1
2-wire input/output connections



4-wire connections

Figure 2-2 shows basic 4-wire signal connections. In this case, the OUTPUT F (force) and S (Kelvin sense) terminals are connected to the thermoelectric cooler, while the INPUT F and S terminals are connected to the temperature sensor. 4-wire sensor connections are recommended for RTD and thermistor sensors with <1kΩ resistance. This type of connection will eliminate lead resistance errors. (See "Sensing considerations" later in this section.)

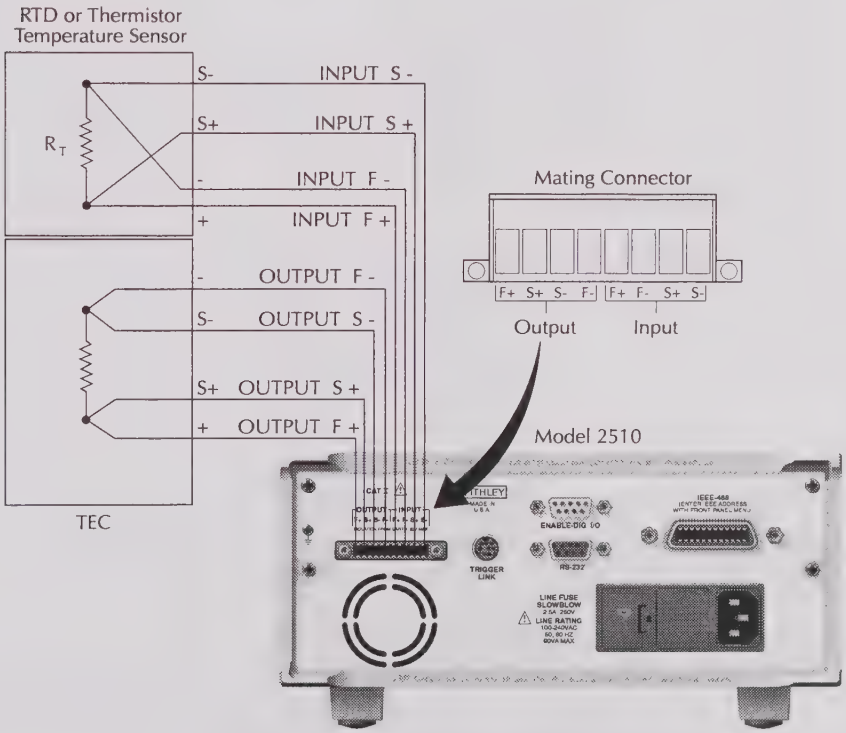


Figure 2-2
4-wire input/output connections

Sense selection

4-wire connections minimize the detrimental effects of lead resistance by sourcing the test current through one set of wires (OUTPUT and INPUT F terminals) and measuring the voltage across the device through a second set of wires (S terminals).

OUTPUT sensing

Sensing though the OUTPUT terminals is automatic with no user programming required. To use sensing with OUTPUT connections, simply make 4-wire connections to the thermoelectric cooler as shown in Figure 2-2. Note that it is recommended that you use 4-wire OUTPUT connections whenever the thermoelectric cooler voltage or resistance must be accurately known.

INPUT sensing

4-wire sensing may be used for both RTD and thermistor sensors, but sense selection is not automatic. The sense mode must be manually set when choosing a sensor type with the CONFIG T or CONFIG R menus. See "Configuring functions" in Section 3 for details.

NOTE *4-wire connections and 4-wire sensing should be used with RTD and thermistor sensors with resistances <1kΩ to avoid errors caused by lead resistance.*

Sensing considerations

Figure 2-3 shows a comparison of 2- and 4-wire sensing methods. In these examples, the INPUT terminals are connected to a thermistor type temperature sensor, but the considerations for RTD sensors and OUTPUT connections to the TEC are similar.

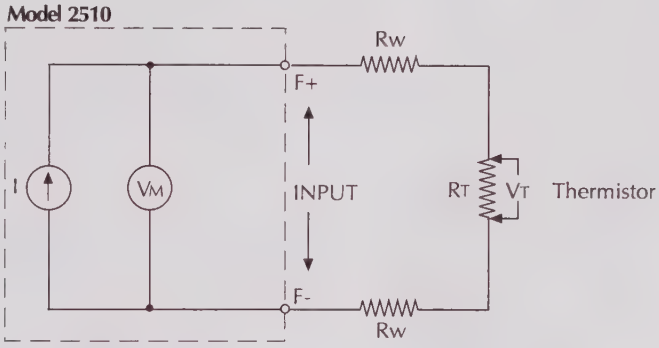
2-wire sensing

The 2-wire connecting method is shown in Figure 2-3A. To determine the sensor resistance, the test current (I) is forced through the F+ and F- test leads and the resistance of the thermistor being measured (R_T). The Model 2510 then measures the voltage across the thermistor through the same F+ and F- connecting wires and computes the thermistor resistance from the measured voltage (V_M) and the current as follows:

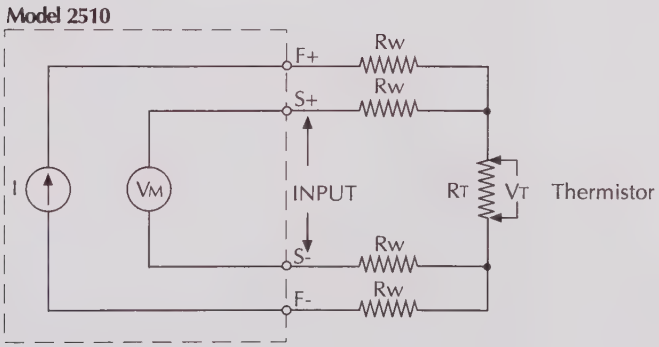
$$R_T = \frac{V_M}{I}$$

The main problem with 2-wire connections is the wire resistance (R_W). Since the test current (I) causes a small but significant voltage drop across the wire resistances, the measured voltage (V_M) will not be exactly the same as the voltage (V_T) directly across the thermistor, and considerable resistance and temperature measurement error can result.

Figure 2-3
Sensing methods



A. 2-Wire Connections



B. 4-Wire Connections

4-wire sensing

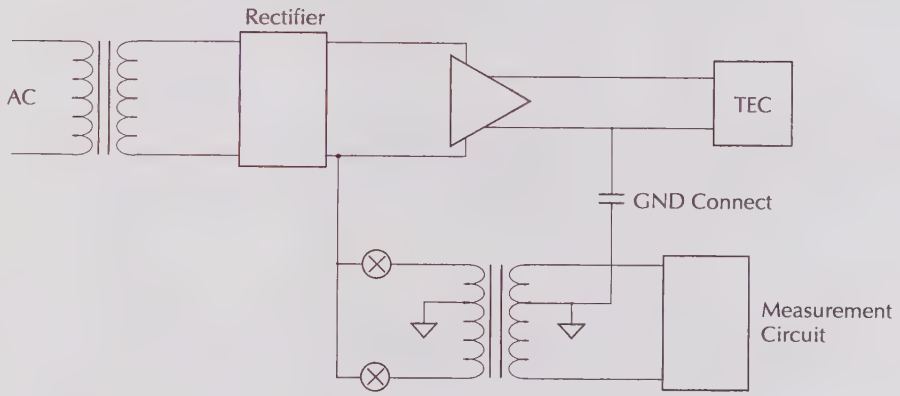
Due to the limitations of the 2-wire method, the 4-wire connection method shown in Figure 2-3B is recommended for sensors <1kΩ. With this configuration, the test current (I) is forced through the thermistor through the F+ and F- wires, while the voltage across the thermistor is measured through a second set of wires connected to the S+ and S- (sense) terminals. Although some small current may flow through the sense wires, it is usually negligible and can generally be ignored for all practical purposes. Since the voltage drop across the sense wires is negligible, the voltage measured by the Model 2510 is essentially the same as the voltage across the thermistor ($V_M \cong V_T$), and more accurate resistance and temperature measurements result from the following resistance calculation:

$$R_T = \frac{V_T}{I}$$

Ground connect mode

Ground connect connects output low to sensor measurement ground, as shown in Figure 2-4. Without ground connect, the output can float up to ±10V from sensor measurement ground.

Figure 2-4
Ground connect mode

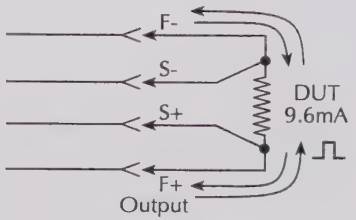


AC ohms measurement

The AC ohms measurement can be used to determine the AC resistance across the OUTPUT F+ and F- terminals. This measurement is typically used as a pre-test of Peltier static resistance before using PID control. Note that the AC ohms measurement is not available from the front panel, but it can be obtained via remote by using the :MEAS:RES:AC? query. See Section 9 for details.

As shown in Figure 2-5, the AC ohms command results in a 9.6mA current being pulsed through the DUT in opposite directions so as to eliminate system voltage offset and minimize temperature change of the Peltier. During thermoelectric (PID) control, dynamic (DC) resistance can be displayed on the front panel display by pressing the DISPLAY TOGGLE key or via remote with the :MEAS:RES:DC? query (Section 9).

Figure 2-5
AC ohms measurement



3
Basic Operation

- **Safety precautions** — Lists safety precautions that should be observed when using the unit.
- **Operation overview** — Summarizes the basic operating characteristics of the Model 2510.
- **Configuring functions** — Provides step-by-step procedures for configuring the Model 2510 temperature, voltage, current, and resistance functions.
- **Adjusting setpoints** — Provides basic information on adjusting setpoints and allowable setpoint ranges.
- **Basic front panel control-measure procedure** — Describes the basic procedure for setting up unit control and measurement operating modes and using the unit from the front panel.
- **Basic remote control-measure procedure** — Describes the basic procedure for setting up unit control and measurement operating modes and using the unit using remote commands.
- **Protection limits** — Discusses the protection limits for the various functions.
- **Setpoint tolerances** — Details using the setpoint tolerance to determine when the control setting is within the desired setpoint window.
- **Temperature sensors** — Covers various aspects of temperature sensors.

Safety precautions

WARNING **The device under test may be at temperature extremes. Improper use of the temperature controller may cause personal injury. Allow sufficient time for the device under test to return to a safe temperature after testing before making personal contact. Use a heat sink or physical barriers as necessary to avoid contact.**

WARNING **The maximum common-mode voltage is 30V DC. Exceeding this value will result in a shock hazard because of the floating source.**

CAUTION **The maximum voltage between the INPUT/OUTPUT S (sense) terminals is 1V. (Difference between force and sense lines for both INPUT and OUTPUT terminals.) Exceeding this value may result in instrument damage.**

Operation overview

Control characteristics

The Model 2510 has the following basic control characteristics:

- Control capabilities — Control thermoelectric cooler (TEC) power to maintain constant temperature, current, voltage, or thermistor resistance.
- Output range — ±10V DC at up to ±5A DC.
- Setpoint range — -50°C to 225°C.
- Setpoint resolution — ±0.001°C.
- Over temperature limit — 250°C maximum.
- Under temperature limit — -50°C minimum.
- Current limit — 1A to 5.25A.

Measurement characteristics

Basic Model 2510 measurement characteristics include:

- Measurement functions — (AC resistance available only via remote.) DC resistance; operating voltage, current, and power; and AC resistance.
- Thermal feedback element types — PT385, PT3916, PT100, D10, F100 RTD; 100Ω, 1kΩ, 10kΩ, and 100kΩ thermistor; AD590 (current) and LM335 (voltage) solid-state temperature sensors.

Limit characteristics

For each control function (T, V, I, and R), you can program limits to protect the DUT for damage or even to perform limit testing. The Model 2510 will display an appropriate status message if a programmed limit has been exceeded, and you can query if a limit has been tripped with an appropriate remote command (see Section 10). See "Configuring functions" in this section for details on programming protection limits for the various functions.

Configuring functions

The following paragraphs list step-by-step procedures for configuring the temperature, voltage, current, and resistance functions.

NOTE *See "PID control concepts" in Section 4 for information on setting PID control loop proportional, integral, and derivative constants. See also "Sensor coefficients" in Section 4 for information on thermistor and RTD coefficients.*

Configuring temperature

Using Table 3-1 as a guide, configure the temperature function as follows:

1. Press CONFIG then T to access the temperature configuration menu.
2. Select PROTECTION, then press ENTER.
3. Select CONTROL, then press ENTER.
4. Choose ENABLE, then press ENTER.
5. Select LOLIM, then press ENTER. Set the low temperature limit using the EDIT keys, then press ENTER again.
6. Choose HILIM, then press ENTER, set the high temperature limit using the EDIT keys, then again press ENTER.
7. Press EXIT, choose SENSOR-TYPE, then press ENTER.
8. Choose the desired temperature sensor (THERMISTOR, RTD, I-SS, or V-SS), then press ENTER.
9. Program the settings for the selected temperature sensor. For example, for an RTD sensor, choose the following:
   - TYPE: Select PT385, PT3916, PT100, D100, F100, or USER type. (For USER type, set α, β, and δ.)
   - I-SRC: Set RTD current (AUTO, 3.3µA, 10µA, 33.3µA, 100µA, 833.3µA, or 2.5mA).
   - SENSE-MODE: Choose 2-WIRE or 4-WIRE sensing. (Use 4-wire for sensors <1kΩ).

10. Press EXIT as needed to return to the CONFIG TEMPERATURE menu.
11. Select UNITS, press ENTER, then select the desired units (°C, °F, or Kelvin).
12. Choose PID, then press ENTER.
13. Program your PROPORTIONAL, INTEGRAL, and DERIVATIVE constants as required.
14. After programming, press EXIT as necessary to return to normal display.

Table 3-1
Temperature configuration menu

Configuration menu item	Description
PROTECTION	Configure temperature protection.
CONTROL	ENABLE or DISABLE temperature protection.
LOLIM	Program lower limit (-50 to +250°C; 223.15 to 523.15K; -58 to +482°F).*
HILIM	Program higher limit (-50 to +250°C; 223.15 to 523.15K; -58 to +482°F).*
SENSOR-TYPE	Select sensor type and settings.
THERMISTOR	Program thermistor settings.
RANGE	Set thermistor range (100Ω, 1kΩ, 10kΩ, 100kΩ).
A	Program A coefficient (-10 to +10).
B	Program B coefficient (-10 to +10).
C	Program C coefficient (-10 to +10).
I-SRC	Set thermistor current (AUTO, 3.3µA, 10µA, 33.3µA, 100µA, 833.3µA, 2.5mA).
SENSE-MODE	Select 2-WIRE or 4-WIRE sense mode.
RTD	Program RTD sensor parameters.
TYPE	Select PT385, PT3916, PT100, D100, F100, USER type. Program USER α, β, δ.
α	Program USER alpha (0 to 0.01).
β	Program USER beta (0 to 1).
δ	Program USER delta (0 to 5).
I-SRC	Set RTD current (AUTO, 3.3µA, 10µA, 33.3µA, 100µA, 833.3µA, 2.5mA).
SENSE-MODE	Select 2-WIRE or 4-WIRE sense mode.
ISS	Program current solid-state sensor settings.
OFFSET	Program offset (K), 0 to 999.999
SCALE-FACTOR	Program slope (K/A), 0 to 9.999e-4
VSS	Program voltage solid-state sensor settings.
OFFSET	Program offset (K), 0 to 999.999
SCALE-FACTOR	Program slope (K/A), 0 to 9.999e-2
UNITS	Select temperature units (°C, Kelvin, °F).
PID	Program PID loop constants for temperature.
PROPORTIONAL	Set proportional constant (0 to 999.999).
INTEGRAL	Set integral constant (0 to 100).
DERIVATIVE	Set derivative constant (0 to 100).

*Lower limit cannot be set higher than higher limit. Higher limit cannot be set lower than lower limit.

Configuring voltage

Using Table 3-2 as a guide, configure the voltage source as follows:

1. Press CONFIG then V to access the voltage configuration menu.
2. Select PROTECTION, then press ENTER.
3. Using the EDIT keys, set the voltage limit to the desired value, then press ENTER.
4. Choose PID, then press ENTER.
5. Program the PROPORTIONAL, INTEGRAL, and DERIVATIVE constants as required.
6. After programming, press EXIT as necessary to return to normal display.

Table 3-2
Voltage source configuration menu

Configuration menu item	Description
PROTECTION	Set voltage protection limit (0.5 to +10.5V).
PID	Program PID loop constants for voltage.
PROPORTIONAL	Set proportional constant (0 to 999.999).
INTEGRAL	Set integral constant (0 to 100).
DERIVATIVE	Set derivative constant (0 to 100).

Configuring current

Using Table 3-3 as a guide, configure the current source as follows:

1. Press CONFIG then I to enter the current configuration menu.
2. Select PROTECTION, then press ENTER.
3. Using the EDIT keys, set the current limit to the desired value, then press ENTER.
4. Choose PID, then press ENTER.
5. Program the PROPORTIONAL, INTEGRAL, and DERIVATIVE constants as required.
6. Press EXIT to return to normal display.

Table 3-3
Current source configuration menu

Configuration menu item	Description
PROTECTION	Set current protection limit (1.00 to 5.25A).
PID	Program PID loop constants for current.
PROPORTIONAL	Set proportional constant (0 to 999.999).
INTEGRAL	Set integral constant (0 to 100).
DERIVATIVE	Set derivative constant (0 to 100).

Configuring resistance

Using Table 3-4 as a guide, configure the resistance function as follows:

1. Press CONFIG then R to access the resistance configuration menu.
2. Select PROTECTION, then press ENTER.
3. Select CONTROL, then press ENTER.
4. Select ENABLE, then press ENTER.
5. Select LOLIM, then press ENTER. Set the low resistance limit to the desired value, then press ENTER.
6. Select HILIM, then press ENTER. Set the high resistance limit to the desired value, then press ENTER.
7. Press EXIT to return to the CONFIG RESISTANCE menu.
8. Select SENSOR-TYPE, then press ENTER.
9. Choose either THERMISTOR or RTD sensor type, then press ENTER.
10. Program the parameters for select sensor. For example, for RTD, set the following:
    - TYPE: Select PT385, PT3916, PT100, D100, F100, or USER type. (For USER type, set α, β, and δ.)
    - I-SRC: Set RTD current (AUTO, 3.3µA, 10µA, 33.3µA, 100µA, 833.3µA, or 2.5mA).
    - SENSE-MODE: Choose 2-WIRE or 4-WIRE sensing. (Use 4-wire for sensors <1kΩ).
11. From the CONFIG RESISTANCE menu, choose PID, then press ENTER.
12. Program the PROPORTIONAL, INTEGRAL, and DERIVATIVE constants as required.
13. After programming, press EXIT as necessary to return to normal display.

Table 3-4
Resistance configuration menu

Configuration menu item	Description
PROTECTION	Configure resistance protection.
CONTROL	ENABLE or DISABLE resistance protection.
LOLIM	Program lower resistance limit: 0.5Ω to 9.999kΩ (RTD or 100Ω thermistor) 5Ω to 99.99kΩ (1kΩ thermistor) 50Ω to 999.9kΩ (10kΩ thermistor) 500Ω to 9.999MΩ (100kΩ thermistor)
HILIM	Program higher resistance limit: 0.5Ω to 9.999kΩ (RTD or 100Ω thermistor) 5Ω to 99.99kΩ (1kΩ thermistor) 50Ω to 999.9kΩ (10kΩ thermistor) 500Ω to 9.999MΩ (100kΩ thermistor)
SENSOR TYPE	Select sensor type and settings.
THERMISTOR	Program thermistor settings.
RANGE	Set thermistor range (100Ω, 1kΩ, 10kΩ, 100kΩ).
I-SRC	Set thermistor current (AUTO, 3.3µA, 10µA, 33.3µA, 100µA, 833.3µA, 2.5mA).
SENSE MODE	Select 2-WIRE or 4-WIRE sense mode.
RTD	Program RTD settings.
I-SRC	Set RTD current (AUTO, 3.3µA, 10µA, 33.3µA, 100µA, 833.3µA, 2.5mA).
SENSE MODE	Select 2-WIRE or 4-WIRE sense mode.
PID	Program PID loop constants for resistance.
PROPORTIONAL	Set proportional constant (0 to 999.999).
INTEGRAL	Set integral constant (0 to 100).
DERIVATIVE	Set derivative constant (0 to 100).

Configuring output

Using Table 3-5 as a guide, configure the output as follows:

1. Press CONFIG then OUTPUT to enter the output configuration menu.
2. Choose GND-CONNECT, then press ENTER.
3. ENABLE or DISABLE ground connect as required, then press ENTER.
4. If you are using the output enable circuit (see Section 5), select ENABLE, then press ENTER. Select ENABLE from the OUTPUT ENABLE menu, then press ENTER again.
5. Press EXIT to return to normal display.

Table 3-5
Output configuration menu

Configuration menu item*	Description
GND-CONNECT	Select input/output analog common (ground) connection.
DISABLE	Disable ground connection.
ENABLE	Enable ground connection.
ENABLE	Enable/disable Digital I/O port output enable line.
DISABLE	Inactivate output enable line.
ENABLE	Activate output enable line.

*See Section 2 for information on reversal and ground connect modes.
See "Output enable line" in Section 5 for output enable information.

Configuring setpoint tolerance

Using Table 3-6 as a guide, follow the steps below to configure setpoint tolerance. See "Setpoint tolerance" later in this section for more details.

1. Press CONFIG, then any one of the four EDIT keys to access the setpoint tolerance menu.
2. From the SETPOINT TOLERANCE MENU, select PERCENTAGE, then press ENTER.
3. Choose PERCENTAGE, then press ENTER.
4. Using the EDIT keys, set the percentage to the desired value, then press ENTER.
5. Choose WINDOW-SIZE, then press ENTER.

6. Use the EDIT keys to set the number of readings to be within the tolerance window, then press ENTER.
7. Press EXIT to return to normal display.

Table 3-6
Setpoint tolerance configuration menu

Configuration menu item	Description
SETPOINT TOLERANCE	
PERCENTAGE	Percent of range for setpoint tolerance (0-100%), default 0.5%.
WINDOW-SIZE	Number readings to be within tolerance (1-100), default 5.

Adjusting setpoints

To adjust a setpoint for a particular control function, simply press the function key for the particular function (T, V, I, or R), then press one of the EDIT keys to enter the setpoint adjustment mode. Use the EDIT keys to adjust the setpoint as follows:

- Use the left and right arrow keys to place the cursor on the digit to be changed.
- Use the ▲ and ▼ keys to increase or decrease the setpoint value.

Table 3-7 summarizes setpoint adjustment ranges for the four control functions.

NOTE *You cannot adjust the setpoint above the upper protection (limit) setting (all functions) or below the lower protection setting (temperature and resistance).*

Table 3-7
Setpoint adjustment ranges

Function	Adjustment range
Temperature (T)* (Constant Peltier temperature)	-50 to +225°C 223.15 to 498.15K -58 to 437°F
Voltage (V)** (Constant Peltier voltage)	-10 to +10V
Current (I)** (Constant Peltier current)	-5 to +5A
Resistance (R)* (Constant thermistor resistance)	0.5Ω to 9.999kΩ (RTD or 100Ω thermistor) 5Ω to 99.99kΩ (1kΩ thermistor) 50Ω to 999.9kΩ (10kΩ thermistor) 500Ω to 9.999MΩ (100kΩ thermistor)

* Setpoint cannot be set above or below protection limits.
** Setpoint cannot be set above protection limit.

Basic front panel control-measure procedure

Use the following procedure to perform the basic control-measure operations.

NOTE *The following procedure assumes that the Model 2510 is already connected to the thermoelectric cooler and temperature sensor as explained in "Input/output connections" in Section 2.*

Step1: Select function.

Press the T,V, I, or R key to select the desired control function:

- T: Temperature function
- V: Voltage function
- I: Current function
- R: Resistance function

Step 2: Configure function.

Press CONFIG then the function key to configure the various operating parameter for the function to be used. For example, to configure the temperature function, press CONFIG then T, and then set the following operating parameters:

- PROTECTION: Enable protection, and program the high and low temperature limits.
- SENSOR-TYPE: Select the type of sensor being used, then set the associated sensor parameters.
- UNITS: Select the desired temperature units.
- PID: Program the proportional, derivative, and integral loop constants.

NOTE *See "Configuring functions" earlier in this section and "Rules to navigate menus" in Section 1 for details on various menu selections and setting parameter values. See "PID control concepts" and "Sensor coefficients" in Section 4 for details on programming the proportional, integral, and derivative constants and sensor parameters.*

Step 3: Adjust setpoint.

Press one of the EDIT keys to enter the setpoint editing mode for the selected function. Use the ▲ and ▼ keys to increment and decrement setpoint values; use the left and right arrow keys to select the cursor position.

Step 4: Set current limit.

If required, set the current limit as needed. (Press CONFIG then I, then use the PROTECTION selection.)

Step 5: Turn output on.

Turn the output on by pressing the ON/OFF OUTPUT key. The OUTPUT indicator will turn on to indicate the output is on.

Step 6: Observe the display.

During operation, you can observe the front panel display for operating and measured parameters. For example, for the temperature function, the display will show the measured temperature and the temperature setpoint, and you can use the DISPLAY TOGGLE key to cycle through the TEC voltage, current, power, and resistance values, as well as the sensor resistance.

Step 7: Turn output off.

When finished, turn the output off by pressing the ON/OFF OUTPUT key. The OUTPUT indicator light will turn off to show the output is off.

Basic remote control-measure procedure

Use the following procedure to perform the basic control-measure operations via remote. See Section 10 for command details and Appendix E for an example program.

NOTE *The following procedure assumes that the Model 2510 is already connected to the thermoelectric cooler and temperature sensor as explained in "Input/output connections" in Section 2.*

Step1: Restore defaults.

Send the following command to restore defaults:

*RST

See Table 1-3 in Section 1.

Step2: Select function.

Send the :SOUR:FUNC command to select the desired control function. For example, to select the temperature function, send the following command:

:SOUR:FUNC TEMP

Step 3: Configure selected function.

Send the commands to program temperature units, protection limits, sensor type, sensor parameters, and PID loop constants. For example, the following commands configure the temperature function:

:UNIT:TEMP CEL	Celsius temperature limits
:SOUR:TEMP:PROT 100	Upper temperature limit = 100°
:SOUR:TEMP:PROT:LOW 10	Lower temperature limit = 10°
:TEMP:TRAN RTD	Select RTD temperature sensor
:TEMP:RTD:TYPE PT100	Select PT100 RTD sensor
:TEMP:CURR:AUTO ON	Use default RTD sensor current
:SYST:RSEN ON	Enable 4-wire sensing
:SOUR:TEMP:LCON 10	Temperature gain constant = 10
:SOUR:TEMP:LCON:INT 0.5	Temperature integral constant = 0.5
:SOUR:TEMP:LCON:DER 0	Temperature derivative constant = 0

Step 4: Program setpoint.

Program the setpoint for the selected control function. For example, the following command programs the temperature setpoint to 50°:

:SOUR:TEMP 50

Step 5: Set current limit.

If required, set the current limit with this command:

:SENS:CURR:PROT<Current_Limit>

Step 6: Turn output on.

Turn the output on to begin PID control by sending the following commands:

:OUTP ON

Step 7: Request readings.

During operation, you can request readings with the :FETCH? query. Note that bus reading elements are controlled with the :FORM:ELEM command. You can also request additional reading types with the following queries:

:MEAS:POW?	Query thermoelectric cooler power
:MEAS:CURR?	Query cooler current
:MEAS:TEMP?	Query temperature
:MEAS:RES?	Query cooler DC resistance
:MEAS:VOLT?	Query cooler voltage

Step 8: Turn output off.

When finished, turn the output off and halt PID control by sending:

:OUTP OFF

Protection limits

Protection limits for the four functions are discussed below. Table 3-8 summarizes protection limit characteristics.

NOTE *For temperature and resistance, the upper limit cannot be set below the lower limit and the lower limit cannot be set above the upper limit. Also, the setpoint cannot exceed limit setting(s).*

Table 3-8
Protection limit characteristics

Function	Limit range	Output Inhibited?*	Display message
Temperature**	Lower limit: -50 to +250°C Upper limit: -50 to +250°C	Yes Yes	UNDER-TEMP OVER-TEMP
Voltage	0.5 to 10.5V	No	VLIM
Current	1.0 to 5.25A	No	ILIM
Resistance**	Lower limit: 0.5Ω to 9.999kΩ (RTD or 100Ω thermistor) 5Ω to 99.99kΩ (1kΩ thermistor) 50Ω to 999.9kΩ (10kΩ thermistor) 500Ω to 9.999MΩ (100kΩ thermistor) Upper limit: 0.5Ω to 9.999kΩ (RTD or 100Ω thermistor) 5Ω to 99.99kΩ (1kΩ thermistor) 50Ω to 999.9kΩ (10kΩ thermistor) 500Ω to 9.999MΩ (100kΩ thermistor)	Yes Yes	UNDER-OHMS OVER-OHMS

* Yes indicates that output turns off if limit is exceeded and cannot be turned on when in limit.
** Lower limit cannot be set above upper limit. Upper limit cannot be set below lower limit.

Temperature protection limits

The upper and lower temperature limits have a valid range of -50 to +250°C. If the lower limit is exceeded, "UNDER-TEMP" will be displayed on the front panel, and Bit 1 (UT) of the Measurement Event Register will be set. If the upper limit is exceeded, "OVER-TEMP" will be displayed on the front panel, and Bit 0 (OT) of the Measurement Event Register will be set. You can program the unit to generate an SRQ if either limit is exceeded (see Section 7). Note that OUTPUT ON/OFF is inhibited by exceeding either limit. The output will turn off if either limit is exceeded, and you cannot turn on the output with under- or over-limit conditions.

To program the temperature limits, press CONFIG then T, then select PROTECTION and program the limits as required. See "Configuring temperature" earlier in this section for details.

Voltage protection limit

The voltage limit has a valid range of 0.5 to 10.5V, and the default value is 10.5. If the voltage limit is exceeded, "VLIM EXCEEDED" will be displayed on the front panel. Bit 2 (OV) of the Measurement Event Register will be set, and you can program the unit to generate an SRQ if the voltage limit is exceeded. Note that OUTPUT ON/OFF is not inhibited by exceeding the voltage limit.

To program the voltage limit, press CONFIG then V, then select PROTECTION and program the limit as required. See "Configuring voltage" earlier in this section for details.

Current protection limit

The current limit has a valid range of 1.0A to 5.25A, and the default value is 2.0. If the current limit is exceeded, "ILIM" will be displayed on the front panel. Bit 3 (CO) of the Measurement Event Register will be set, and you can program the unit to generate an SRQ if the current limit is exceeded (see Section 7). Note that OUTPUT ON/OFF is not inhibited by exceeding the current limit.

To program the current limit, press CONFIG then I, then select PROTECTION and program the limit as required. See "Configuring current" earlier in this section for details.

Resistance protection limits

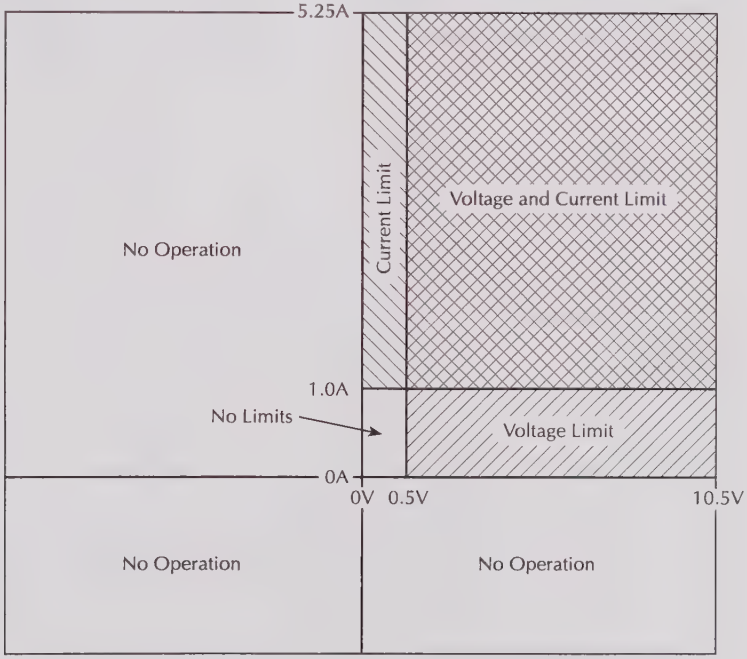
The upper and lower resistance limits apply only to RTD and thermistor sensors (see Table 3-8). If the lower limit is exceeded, "URR" will be displayed, and Bit 5 (UR) of the Measurement Event Register will be set. If the upper limit is exceeded, "ORR" will be displayed, and Bit 4 (OR) of the Measurement Event Register will be set. You can program the unit to generate an SRQ if either limit is exceeded (see Section 7). OUTPUT ON/OFF is inhibited by exceeding either resistance limit.

To program the resistance limits, press CONFIG then R, then select PROTECTION and program the limits as required. See "Configuring resistance" earlier in this section for details.

Voltage and current limit operating boundaries

The combination of the voltage and current limits restricts the allowed operating area, or operating boundaries, as shown in Figure 3-1. In this example, the voltage limit is set to 5V, while the current limit is 2A. The shaded areas of the graph show output currents or voltages that are restricted by the respective limits.

Figure 3-1
Voltage and current limit operating boundaries



Setpoint tolerance

The setpoint tolerance feature allows you to program the Model 2510 to indicate when the setpoint is within the desired range, or window, for a certain number of readings. For example, with a tolerance of 1%, the setpoint will be within tolerance when the temperature is within ±2.75°C of the setpoint for the programmed number of readings.

Ranges

Ranges by function are shown in Table 3-9.

Table 3-9
Setpoint tolerance ranges

Function	Range
Temperature	275°C 275K 495°F
Voltage	20V
Current	10A
Resistance	200kΩ (100kΩ thermistor) 80kΩ (10kΩ thermistor) 10kΩ (1kΩ thermistor) 1kΩ (100Ω thermistor or RTD)

Operation

To configure the setpoint tolerance feature from the front panel, press CONFIG followed by any one of the four EDIT keys (◀, ▲, ▼, ▶). The unit will display the setpoint tolerance configuration menu. (See "Configuring setpoint tolerance" and Table 3-6 earlier in this section.) Use the EDIT, ENTER, and EXIT keys to set the percent tolerance and window size as required.

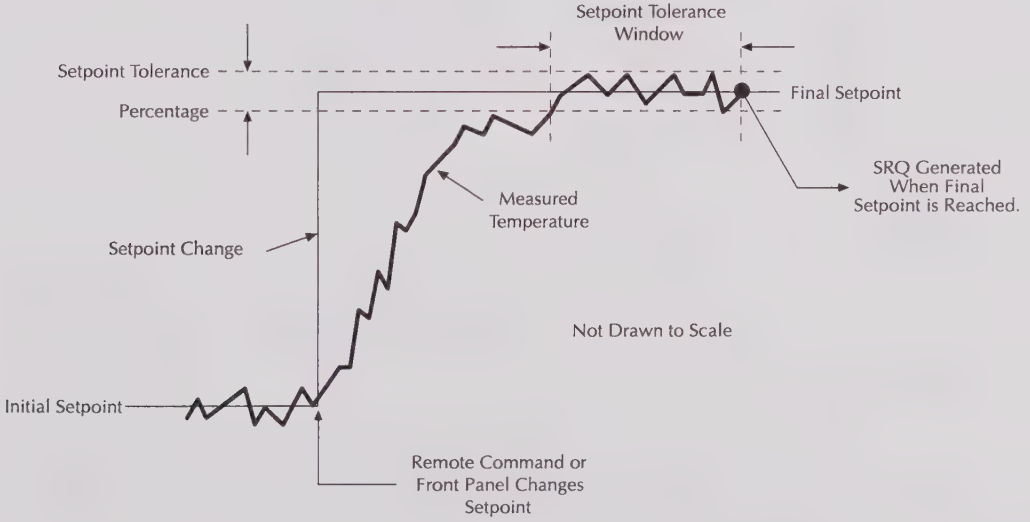
Setpoint tolerance indicator

The front panel "*" annunciator will turn on when the output is within the setpoint tolerance for the programmed number of readings. The unit can also be programmed to generate an SRQ when setpoint tolerance is reached via bit 12 (SPT) of the Measurement Event Register. See Section 7, "Status structure" for details.

Setpoint tolerance operation

Figure 3-2 demonstrates basic setpoint tolerance operation. Operation begins at the Initial Setpoint where the temperature stabilizes at the programmed setpoint. Once the setpoint has been changed (either from the front panel or with a remote command), the temperature gradually increases to the new setpoint as shown. Once the temperature reaches the Setpoint Tolerance Percentage value, it then waits until the number of readings within the Setpoint Tolerance Window have occurred, at which point the Final Setpoint is reached. At that point, the unit issues an SRQ over the bus (if programmed to do so) and turns on the "*" annunciator to indicate the final setpoint has been reached.

Figure 3-2
Setpoint tolerance operation



NOTE *Setpoint tolerance SRQs can be used to generate a temperature sweep. See Appendix E for details and an example program.*

Temperature sensors

Open and shorted lead status indications

For temperature measurements only, Open Lead and Shorted Lead status indications identify when a sensor lead is open or shorted. The front panel will display "Open Lead" or "ShrtLead" where appropriate. The conditions for open and shorted lead are summarized in Table 3-10.

Table 3-10
Sensor open lead and shorted lead ranges

Sensor	Open Lead	Shorted Lead
Thermistor	Temp. < -50°C	Temp. > 250°C
RTD	Temp. > 250°C	Temp. < -50°C
V-SS	Temp. < -50°C	-0.001V to +0.001V
I-SS	Temp. < -50°C	Temp. > 250°C

Solid-state sensor restrictions

I-SS and V-SS temperature sensors cannot be selected with the resistance function active. The resistance function is intended for use only with thermistor and RTD sensors.

Thermistor sensor ranges

Available thermistor sensor ranges are 100Ω, 1kΩ, 10kΩ, and 100kΩ. For optimum accuracy, use the appropriate range for your sensor. The Model 2510 will display "URR" (Under Resistance Range) or "ORR" (Over Resistance Range) if the sensor resistance is outside the optimum range.

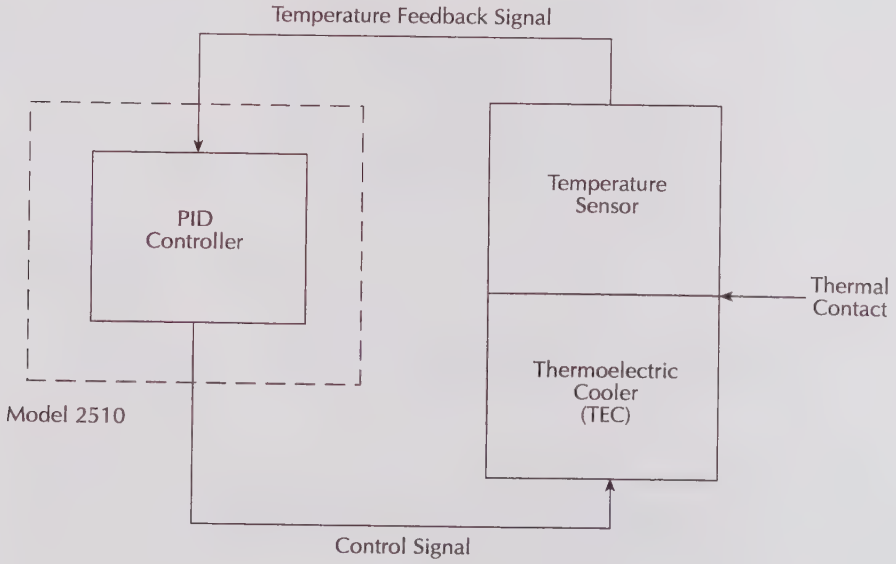
4
PID Control Concepts

- **Temperature control model** — Briefly covers the model for temperature control.
- **Temperature control methods** — Briefly covers various methods used for temperature control, including on-off, proportional, proportional-derivative, and proportional-integral derivative.
- **PID tuning** — Provides a basic step-by-step procedure for tuning the PID loop for optimum operation.
- **Sensor coefficients** — Covers thermistor and RTD temperature coefficients.

Temperature control model

Figure 4-1 shows the overall control model for the system. The Model 2510 provides current to the thermoelectric cooler (TEC), which uses the Peltier effect to maintain temperature at the desired setpoint. When current is forced through the TEC in one direction, the device heats; when current is forced through the TEC in the opposite direction, the device cools. In this manner, temperature is maintained at the setpoint, and equilibrium can be maintained. Temperature feedback for the PID controller is provided by the temperature sensor, which is in thermal contact with the thermoelectric cooler. PID control parameters can be individually set to assure optimum temperature control as outlined below.

Figure 4-1
Temperature control model



Temperature control methods

The following discussion outlines comparisons between various temperature control methods. Terms for this discussion are defined as follows:

- T_S = setpoint temperature
- T_C = cooler temperature
- W = power applied to thermoelectric cooler.

On-off control

The on-off control method, used by many conventional thermostats, is inadequate for thermoelectric cooler (TEC) control situations. When the TEC is warmer than the setpoint temperature, the TEC power is turned on to maximum. Once the TEC is cooler than the setpoint temperature, the TEC power is turned off completely. The turn-on and turn-off temperatures usually differ by a small amount to minimize the amount of temperature cycling, resulting in a certain amount of hysteresis. However, temperature fluctuations are generally too large for precise TEC temperature control due in large part to the thermal time constant of the TEC and the hysteresis required to limit excessive temperature cycling.

Proportional (P) control

A proportional controller usually performs better than the on-off type discussed above by applying power, W, to the cooler in proportion to the difference in temperature between the cooler and setpoint temperatures as follows:

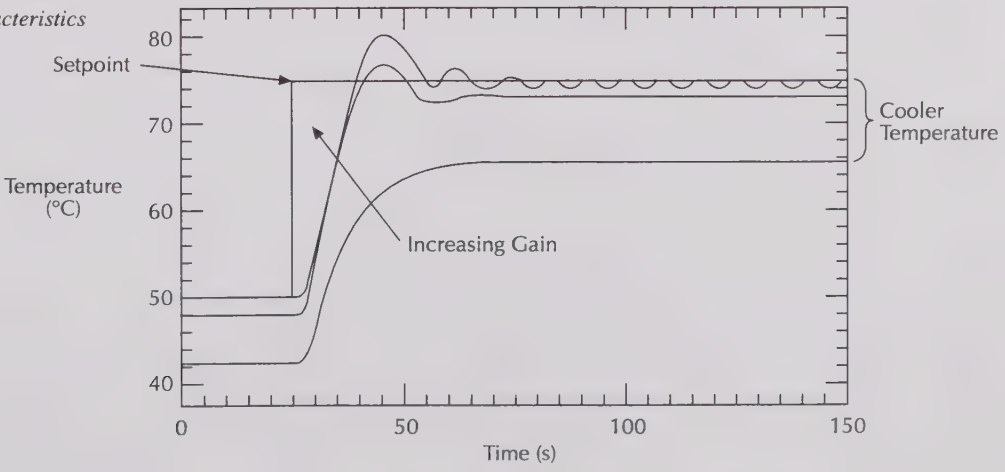
$$W = K_p \times (T_c - T_s)$$

Here, K_p is the proportional gain of the controller.

As its gain is increased, the system responds faster to changes in setpoint, but it becomes progressively underdamped and eventually unstable, as shown in Figure 4-2. The final cooler temperature lies above the setpoint with this method because some temperature difference is required to maintain power to the cooler.

To use the Model 2510 as a P controller, set the derivative and integral constants to zero.

Figure 4-2
Proportional control characteristics



Proportional-derivative (PD) control

The stability and overshoot problems that occur when a proportional controller is used at high gain can be minimized by adding a term proportional to the time-derivative of the error signal as follows:

$$W = K_p \times \left[(T_C - T_S) + K_d \times \frac{d}{dt}(T_C - T_S) \right]$$

Here, K_d is the derivative gain.

Proportional-integral-derivative (PID) control

Although PD control deals well with the overshoot and ringing problems associated with proportional control, it does not cure the problem with steady-state error. However, it is possible to eliminate this error while using relatively low proportional gain by adding an integral term to the control function as follows:

$$W = K_p \times \left[(T_C - T_S) + K_d \times \frac{d}{dt}(T_C - T_S) + K_i \times \int (T_C - T_S)dt \right]$$

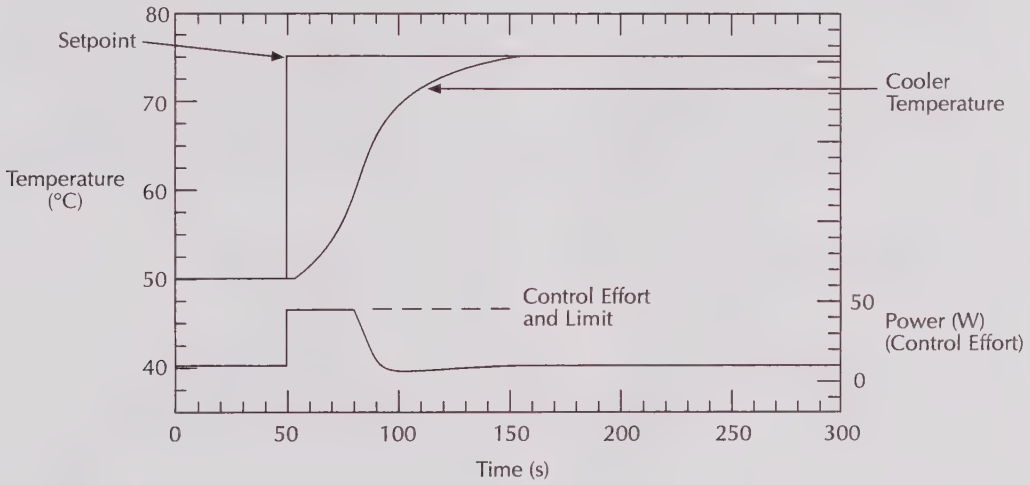
Here, K_i, the integral gain parameter, is sometimes known as the controller reset level.

This form of function is known as proportional-integral-differential, or PID control for which the Model 2510 is optimized. The effect of the integral term is to change the TEC power until the time-averaged value of the temperature error is zero. Figure 4-3 shows the result of adding the integral term to the control function.

To summarize:

- K_p, the proportional constant, controls the loop gain.
- K_i, the integral constant, offsets errors.
- K_d, the derivative constant, affects the damping rate.

Figure 4-3
PID control characteristics



PID tuning

It will be necessary for you to tune the Model 2510 PID loop controller by trial and error. The method for tuning the PID loop outlined below is intended only as a starting point, and some experimentation may be required for optimum performance. This procedure is based on the assumption that a critically damped system is optimal, and that stability and noise performance must be traded for response time. See "Configuring functions" in Section 3 for information on setting up functions including programming the proportional, integral, and derivative constants covered in these procedures.

CAUTION **This procedure may result in large temperature oscillations during the tuning process that could cause damage to the thermoelectric cooler.**

1. Adjust the temperature setpoint value (T_S) to a typical value for thermoelectric cooler, and set the Derivative (D) and Integral (I) constants to 0.
2. Select a safe value for the maximum TEC power, and increase the proportional gain (P) until the system is just oscillating. Note the period of oscillation, then reduce the gain by 30%. Note, however, that if you adjust for maximum power (control effort limit) and tune for that setting, specific PID parameters will result. If you use those same PID values for a different limit, the system may respond differently.

3. Suddenly decreasing or increasing T_S by about 5% should induce underdamped oscillations. Try several Derivative values, and choose a value that gives a critically damped response. Typically, D will be about one third of the oscillation period obtained in step 2 above.
4. Slowly increase the Integral (I) value until oscillation just begins, then reduce this level by a factor of two or three. This reduction should be enough to stop the oscillation. Use the lowest I value that gives adequate performance.
5. Check that the overall temperature control characteristics of the Model 2510 using these settings is satisfactory under the conditions it will be used.

Sensor coefficients

Thermistor coefficients

The thermistor sensor coefficients A, B, and C linearize the thermistor temperature-resistance curve and are related using the Steinhart and Hart equation as follows:

$$\frac{1}{T} = A + B[\ln R] + C[\ln R]^3$$

Where:
T = Temperature (K)
R = Thermistor resistance (ohms)
A, B, C = Thermistor sensor coefficients

These coefficients should be set to values specified by the thermistor manufacturer. See "Configuring temperature" in Section 3 for programming details.

RTD coefficients

In a similar manner, RTD sensor coefficients α, β, and δ linearize the RTD temperature-resistance curve using the Callendar-Van Dusen equation in the following manner:

$$R_T = R_0 + R_0\alpha\left[T - \delta\left(\frac{T}{100} - 1\right)\left(\frac{T}{100}\right) - \beta\left(\frac{T}{100} - 1\right)\left(\frac{T}{100}\right)^3\right]$$

Where:
R_T = Resistance at temperature T (°C)
R_0 = Resistance at temperature 0°C
α = Temperature coefficient at 0°C (typically 0.00392Ω/°C)
δ = 1.49 (typical for 0.00392 platinum RTD)
β = 0 for T >0°; 0.11 (typical) for T<0°

These coefficients should be set to values specified by the RTD manufacturer. See "Configuring temperature" in Section 3 for programming details.

5
Digital I/O Port and Output Enable

- **Digital I/O port** — Details Digital I/O port configuration, connections, and setting I/O port operating parameters.
- **Output enable line** — Discusses output enable line operation, and describes how to activate the line.

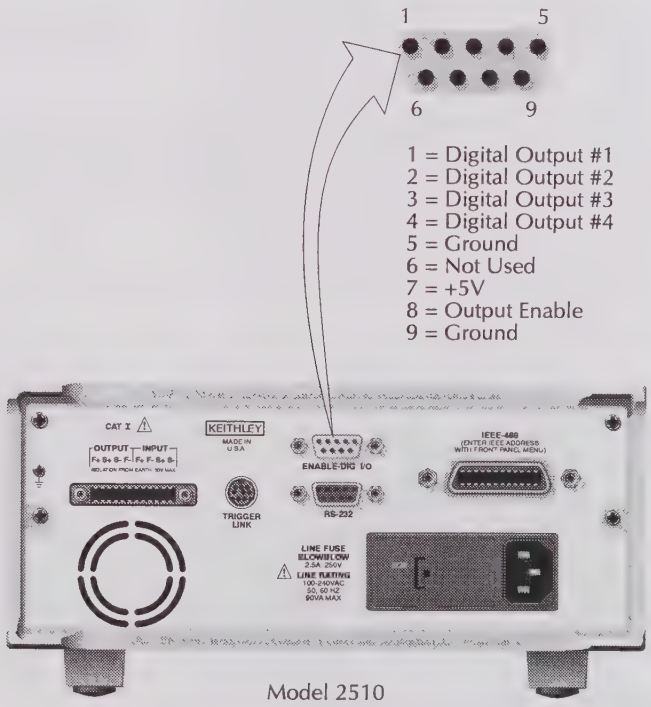
Digital I/O port

The Model 2510 has a digital input/output port that can be used to control external digital circuitry and for a test fixture output enable circuit.

Port configuration

The Digital I/O Port (ENABLE - DIG I/O connector) is located on the rear panel and is shown in Figure 5-1. Note that a standard male DB-9 connector is used.

Figure 5-1
Digital I/O port



Digital output lines

The port provides four output lines. Each open-collector output can be set high (+5V) or low (0V). Each output line can source up to 2mA or all four lines together can sink up to 500mA.

Output enable line

The output enable line is intended for use with a test fixture to disable the output when the protection lid is open. See "Output enable line" later in this section for details.

+5V output

The Digital I/O Port provides a +5V output that can be used to drive external logic circuitry. Maximum current output for this line is 600mA. This line is protected by a self-resetting fuse (one hour recovery time).

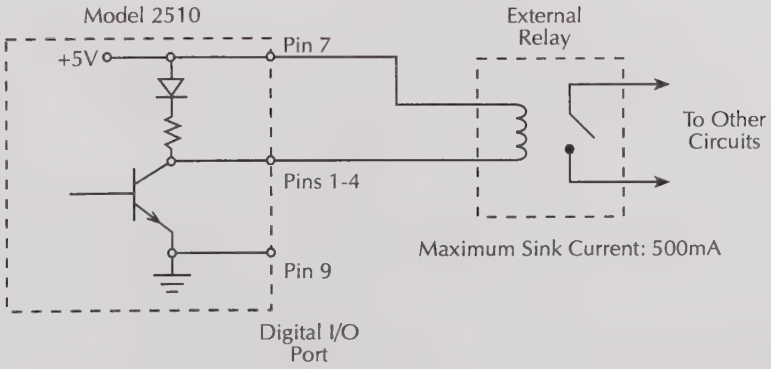
Digital output configuration

There are two basic methods to connect external components to the digital output lines, sink operation, and source operation.

Sink operation

Figure 5-2 shows the basic output configuration for sink operation. Note that the external relay coil is connected between the digital output line (pins 1 to 4) and +5V (pin 7). With this configuration, the digital output line must be set LO to energize the relay, and the maximum sink current is 500mA.

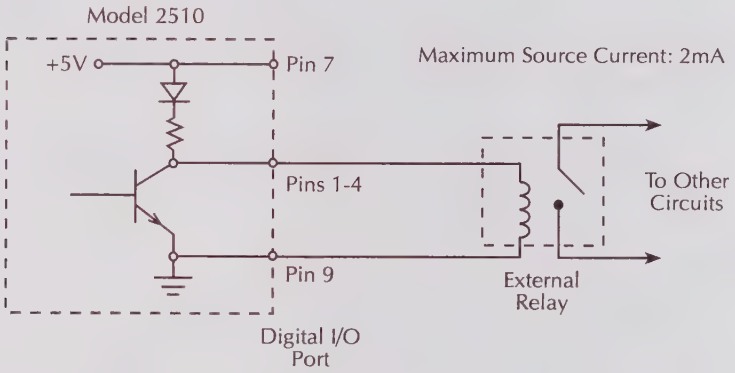
Figure 5-2
Sink operation



Source operation

Figure 5-3 shows the basic output configuration for source operation. In this case, the external relay coil is connected between the digital output line (pins 1 to 4) and ground (pin 9). With this configuration, the digital output line must be set HI to energize the relay, and the maximum source current is 2mA.

Figure 5-3
Source operation



Controlling digital output lines

Set digital output line logic levels from the front panel as follows:

1. Press the MENU key.
2. Select GENERAL, then press ENTER.
3. Select DIGOUT, then press ENTER.
4. Using the EDIT keys, set the digital output parameter to the desired decimal value (see Table 5-1), then press ENTER. For example, to set the output lines to L, H, H, H, set the digital output parameter value to 7, then press ENTER.
5. Press EXIT to return to normal display.

Table 5-1
Digital output line settings

OUT 4	OUT 3	OUT 2	OUT 1	Decimal value*
L	L	L	L	0
L	L	L	H	1
L	L	H	L	2
L	L	H	H	3
L	H	L	L	4
L	H	L	H	5
L	H	H	L	6
L	H	H	H	7
H	L	L	L	8
H	L	L	H	9
H	L	H	L	10
H	L	H	H	11
H	H	L	L	12
H	H	L	H	13
H	H	H	L	14
H	H	H	H	15

L = Low (Gnd)
H = High (>+3V)
*Via remote, send :SOUR2:TTL<Decimal_value>. See :SOURce2 subsystem in Section 10.

Output enable line

The Digital I/O Port provides an output enable line for use with a test fixture activate switch. When properly used, the OUTPUT of the Model 2510 will turn OFF when the lid of the test fixture is opened.

Overview

When the output enable line is active (see below), the output of the Model 2510 cannot be turned on unless the enable line is pulled low through a switch to ground as shown in Figure 5-4A. If the lid of the test fixture opens (Figure 5-4B), the switch opens, and the enable line goes high, turning the OUTPUT of the Model 2510 OFF. The output can only be turned back on by first closing the lid of the test fixture and then pressing the OUTPUT ON/OFF key.

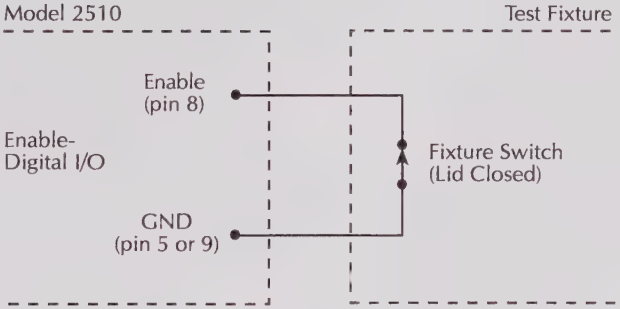
WARNING **The Model 2510 TEC SourceMeter was designed as a general-purpose instrument, which precludes the use of voltage and/or current barriers. Therefore, there is no intrinsically safe barrier, including output enable, in the Model 2510. Even with the output enable feature in use, it is possible under certain fault conditions (for example with a programming error or instrument failure) to present a shock hazard to the operator or damage devices even when the system appears to be safe (such as when voltage and/or current conditions are below safety limits or the output is disabled.) Use appropriate guards, shields, and insulators on external connectors to protect any user or maintenance personnel from the risk of electric shock or mechanical injury.**

Activating output enable

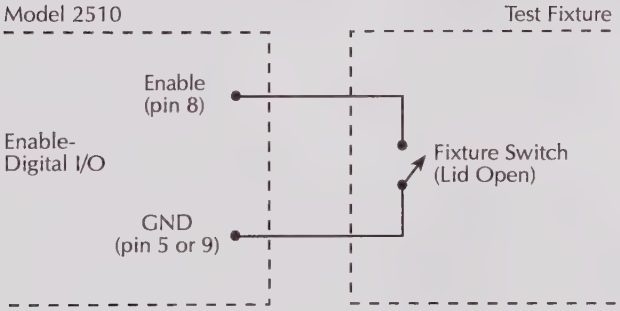
Output enable can be activated by using the CONFIG OUTPUT menu. Follow these steps to activate output enable:

1. Press CONFIG then OUTPUT.
2. Select ENABLE, then press ENTER.
3. From the OUTPUT ENABLE menu, choose ENABLE, then press ENTER to activate the output enable line.
4. Press EXIT to return to normal display.

Figure 5-4
Using output enable



A. 2510 OUTPUT can be turned on.



B. 2510 OUTPUT turns off and cannot be re-enabled until closed again.

6
Remote Operations

- **Differences: remote vs. local operation** — Summarizes remote operation enhancements and local-to-remote and remote-to-local transitions.
- **Selecting an interface** — Describes how to select between the GPIB and RS-232 interfaces.
- **GPIB operation** — Covers GPIB bus standards, bus connections, and primary address selection.
- **RS-232 interface operation** — Outlines use of the RS-232 interface to control the Model 2510 via remote.
- **Front panel GPIB operation** — Summarizes GPIB error messages, status indicators, and using the LOCAL key.
- **General bus commands** — Describes general bus commands used for fundamental GPIB control.
- **Programming syntax** — Describes the basic programming syntax for both common commands (Section 8) and SCPI commands (Section 10).

Differences: remote vs. local operation

Local-to-remote transition

When changing from Local to Remote operation, the following takes place:

- All menus are exited.
- All pending front panel commands are aborted.
- All other settings remain unaffected.

Remote-to-local transition

When changing from Remote to Local operation, the following takes place:

- The Model 2510 stops taking readings and is placed into idle.
- All user-defined displays are disabled.
- Readings are continuously taken.

Setting interface parameters

- RS-232 settings for Baud Rate, Parity, and Data Bits are accessible only from the front panel while RS-232 is enabled.
- The IEEE-488 address can only be changed from the front panel while the IEEE-488 bus is enabled.

Selecting an interface

The Model 2510 supports two built-in remote interfaces:

- GPIB (General Purpose Interface Bus)
- RS-232 interface

You can use only one interface at a time. The factory default interface selection is the GPIB bus. You can select the interface only from the front panel. The interface selection is stored in non-volatile memory; it does not change when power has been off or after a remote interface reset.

The GPIB bus is the IEEE-488 interface. You must select a unique address for the Model 2510. The address is displayed when the instrument is turned on, and you can use the MENU/COMMUNICATION/GPIB selection to view the address after the unit is turned on. At the factory, the address is set to 15.

The RS-232 interface is a serial interface. Programmable aspects of this interface include the following (factory default settings are shown in parentheses):

- Baud rate (9600)
- Data bits (8)
- Parity (none)
- Terminator (CR)
- Flow control (none)

An interface is selected and configured from the COMMUNICATION option of the Main Menu (see "Main menu" in Section 1). For details on the programmable aspects of the interfaces, see "Primary address and RS-232 interface operation" in this section.

NOTE *When changing interface selections, the Model 2510 performs a power-on reset. To check and/or change options of the selected interface, you must re-enter the menu structure.*

GPIB operation

This section contains information about GPIB standards, bus connections, and primary address options. See Appendix C for more GPIB information.

GPIB standards

The GPIB is the IEEE-488 instrumentation data bus with hardware and programming standards originally adopted by the IEEE (Institute of Electrical and Electronic Engineers) in 1975. The Model 2510 conforms to these standards:

- IEEE-488.1-1987
- IEEE-488.2-1992

The above standards define a syntax for sending data to and from instruments, how an instrument interprets this data, what registers should exist to record the state of the instrument, and a group of common commands. The Model 2510 also conforms to this standard:

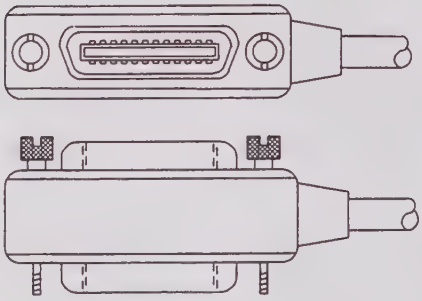
- SCPI 1996.0 (Standard Commands for Programmable Instruments)

This standard defines a command language protocol. It goes one step farther than IEEE-488.2-1992 and defines a standard set of commands to control every programmable aspect of an instrument.

GPIB connections

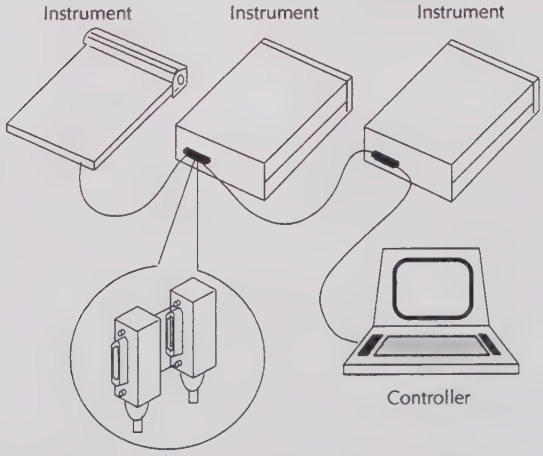
To connect the Model 2510 to the GPIB bus, use a cable equipped with standard IEEE-488 connectors as shown in Figure 6-1.

Figure 6-1
IEEE-488 connector



To allow many parallel connections to one instrument, stack the connectors. Two screws are located on each connector to ensure that connections remain secure. Figure 6-2 shows a typical connecting scheme for a multi-unit test system.

Figure 6-2
IEEE-488 connections



To avoid possible mechanical damage, stack no more than three connectors on any one unit.

NOTE *To minimize interference caused by electromagnetic radiation, use only shielded IEEE-488 cables. Available shielded cables from Keithley are Models 7007-1 and 7007-2.*

To connect the Model 2510 to the IEEE-488 bus, follow these steps:

1. Line up the cable connector with the connector located on the rear panel. The connector is designed so it will fit only one way. Figure 6-3 shows the location of the IEEE-488 connector.
2. Tighten the screws securely, making sure not to overtighten them.
3. Connect any additional connectors from other instruments as required for your application.
4. Make sure the other end of the cable is properly connected to the controller. Most controllers are equipped with an IEEE-488 style connector, but a few may require a different type of connecting cable. See your controller's instruction manual for information about properly connecting to the IEEE-488 bus.

NOTE *You can only have 15 devices connected to an IEEE-488 bus, including the controller. The maximum cable length is either 20 meters or two meters multiplied by the number of devices, whichever is less. Not observing these limits may cause erratic bus operation.*

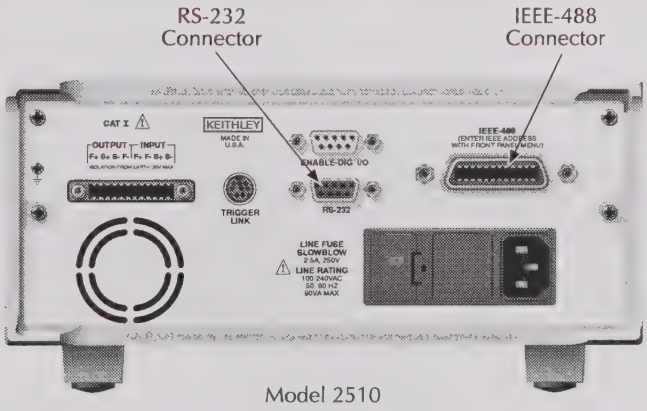


Figure 6-3
IEEE-488 and RS-232 connector locations

Primary address

The Model 2510 ships from the factory with a GPIB primary address of 15. When the unit powers up, it momentarily displays the primary address. You can set the address to a value from 0 to 30, but do not assign the same address to another device or to a controller that is on the same GPIB bus (controller addresses are usually 0 or 21).

The primary address can be checked and/or changed from the COMMUNICATION option of the Main Menu. (See "Main menu" in Section 1.)

RS-232 interface operation

NOTE *The programmable aspects of RS-232 operation (baud rate, data bits, parity, and terminator are configured from the COMMUNICATION option of the Main Menu. (See "Main menu" in Section 1.)*

Sending and receiving data

The RS-232 interface transfers data using 8 data bits, 1 stop bit, and no parity. Make sure the device you connect to the Model 2510 also uses these settings.

You can break data transmissions by sending a ^C (decimal 3) or ^X (decimal 18) character string to the instrument. This clears any pending operation and discards any pending output.

Baud rate

The baud rate is the rate at which the Model 2510 and the programming terminal communicate. Choose one of these available rates:

- 57600
- 38400
- 19200
- 9600
- 4800
- 2400
- 1200
- 600
- 300

The factory selected baud rate is 9600.

When you choose a baud rate, make sure the programming terminal or printer that you are connecting to the Model 2510 can support the baud rate you selected. Both the Model 2510 and the other device must be configured for the same baud rate.

Data bits and parity

The RS-232 interface can be configured to send/receive data that is 7 or 8 bits long using even, odd, or no parity. No parity is only valid when using 8 data bits.

Terminator

The Model 2510 can be configured to terminate each program message that it transmits to the controller with any of the following combinations of <CR> and <LF>:

<CR>	Carriage return
<CR+LF>	Carriage return and line feed
<LF>	Line feed
<LF+CR>	Line feed and carriage return

Flow control (signal handshaking)

Signal handshaking between the controller and the instrument lets the two devices communicate with each other about readiness to receive data. The Model 2510 does not support hardware handshaking (flow control).

Software flow control is in the form of XON and XOFF characters and is enabled when XON-XOFF is selected from the RS-232 FLOW CONTROL menu. When the input queue of the unit becomes more than ¾ full, the instrument issues an XOFF command. The control program should respond to this and stop sending characters until the Model 2510 issues the XON, which it will do once its input buffer has dropped below half-full. The Model 2510 recognizes XON and XOFF sent from the controller. An XOFF will cause the instrument to stop outputting characters until it sees an XON. Incoming commands are processed after the <CR> character is received from the controller.

If NONE is the selected flow control, there will be no signal handshaking between the controller and the Model 2510. Data will be lost if transmitted before the receiving device is ready.

RS-232 connections

The RS-232 serial port is connected to the serial port of a computer using a straight-through RS-232 cable terminated with DB-9 connectors. Do not use a null modem cable. The serial port uses the transmit (TXD), receive (RXD), and signal ground (GND) lines of the RS-232 standard. Figure 6-4 shows the rear panel connector for the RS-232 interface, and Table 6-1 shows the pinout for the connector. (See Figure 6-3 for the rear panel connector location.)

If your computer uses a DB-25 connector for the RS-232 interface, you will need a cable or adapter with a DB-25 connector on one end and a DB-9 connector on the other, wired straight through (not null modem).

NOTE *To minimize interference caused by electromagnetic radiation, use only shielded RS-232 cables such as the Keithley Model 7009-5.*

Figure 6-4
RS-232 interface connector

Rear Panel Connector

Table 6-1
RS-232 connector pinout

Pin number	Description
1	Not used
2	TXD, transmit data
3	RXD, receive data
4	Not used
5	GND, signal ground
6	Not used
7	RTS, ready to send
8	CTS, clear to send
9	Not used

NOTE: CTS and RTS are tied together.

Table 6-2 provides pinout identification for the 9-pin (DB-9) or 25-pin (DB-25) serial port connector on the computer (PC).

Table 6-2
PC serial port pinout

Signal	DB-9 pin number	DB-25 pin number
DCD, data carrier detect	1	8
RXD, receive data	2	3
TXD, transmit data	3	2
DTR, data terminal ready	4	20
GND, signal ground	5	7
DSR, data set ready	6	6
RTS, request to send	7	4
CTS, clear to send	8	5
RI, ring indicator	9	22

Front panel GPIB operation

This section describes aspects of the front panel that are part of GPIB operation, including messages, status indicators, and the LOCAL key.

Error and status messages

During operation, various error and status messages associated with IEEE-488 programming may be displayed (see Appendix B). The instrument can be programmed to generate an SRQ, and command queries can be performed to check for specific error conditions.

GPIB status indicators

The REM (remote), TALK (talk), LSTN (listen), and SRQ (service request) annunciators show the GPIB bus status. Each of these indicators is described below.

REM

This indicator shows when the instrument is in the remote state. REM does not necessarily indicate the state of the bus REN line, as the instrument must be addressed to listen with REN true before the REM indicator turns on. When the instrument is in remote, all front panel keys, except for the DISPLAY TOGGLE/LOCAL key, are locked out. When REM is turned off, the instrument is in the local state, and front panel operation is restored.

NOTE *If LLO (Local LockOut) is in effect, LOCAL will be locked out. OUTPUT OFF is still operational in remote.*

TALK

This indicator is on when the instrument is in the talker active state. Place the unit in the talk state by addressing it to talk with the correct MTA (My Talk Address) command. TALK is off when the unit is in the talker idle state. Place the unit in the talker idle state by sending an UNT (Untalk) command, addressing it to listen, or sending the IFC (Interface Clear) command.

LSTN

This indicator is on when the Model 2510 is in the listener active state, which is activated by addressing the instrument to listen with the correct MLA (My Listen Address) command. LSTN is off when the unit is in the listener idle state. Place the unit in the listener idle state by sending UNL (Unlisten), addressing it to talk, or sending IFC (Interface Clear) command over the bus.

SRQ

You can program the instrument to generate a service request (SRQ) when one or more errors or conditions occur. When this indicator is on, a service request has been generated. This indicator stays on until the serial poll byte is read or all the conditions that caused SRQ have been cleared. See Section 7, "Status structure" for more information.

DISPLAY TOGGLE/LOCAL key

The DISPLAY TOGGLE/LOCAL key cancels the remote state and restores local operation of the instrument.

Pressing the DISPLAY TOGGLE/LOCAL key also turns off the REM indicator and returns the display to normal if a user-defined message was displayed.

If the LLO (Local Lockout) command is in effect, the DISPLAY TOGGLE/LOCAL key is also inoperative.

For safety reasons, the OUTPUT key will still be active in LLO.

General bus commands

General commands are those commands, such as DCL, that have the same general meaning regardless of the instrument. Table 6-3 lists the general bus commands, which are described in detail in Appendix C.

Table 6-3
General bus commands

Command	Effect on Model 2510
REN	Goes into remote when next addressed to listen.
IFC	Goes into talker and listener idle states.
LLO	LOCAL key locked out.
GTL	Cancel remote; restore Model 2510 front panel operation.
DCL	Returns all devices to known conditions.
SDC	Returns Model 2510 to known conditions.
GET	Initiates a trigger.
SPE, SPD	Serial polls the Model 2510.

REN (remote enable)

The remote enable command is sent to the Model 2510 by the controller to set up the instrument for remote operation. Generally, the instrument should be placed in the remote mode before you attempt to program it over the bus. Setting REN true does not place the instrument in the remote state. You must address the instrument to listen after setting REN true before it goes into remote.

IFC (interface clear)

The IFC command is sent by the controller to place the Model 2510 in the local, talker, listener idle states. The unit responds to the IFC command by cancelling front panel TALK or LSTN lights, if the instrument was previously placed in one of these states.

Note that this command does not affect the status of the instrument. Settings, data, and event registers are not changed.

To send the IFC command, the controller need only set the IFC line true for a minimum of 100µs.

LLO (local lockout)

Use the LLO command to prevent local operation of the instrument. After the unit receives LLO, all of its front panel controls except OUTPUT OFF are inoperative. In this state, pressing LOCAL will not restore control to the front panel. The GTL command restores control to the front panel. Cycling power will also cancel local lockout.

GTL (go to local)

Use the GTL command to put a remote-mode instrument into local mode. The GTL command also restores front panel key operation.

DCL (device clear)

Use the DCL command to clear the GPIB interface and return it to a known state. Note that the DCL command is not an addressed command, so all instruments equipped to implement DCL will do so simultaneously.

When the Model 2510 receives a DCL command, it clears the Input Buffer and Output Queue, cancels deferred commands, and clears any command that prevents the processing of any other device command. A DCL does not affect instrument settings and stored data.

SDC (selective device clear)

The SDC command is an addressed command that performs essentially the same function as the DCL command. However, since each device must be individually addressed, the SDC command provides a method to clear only selected instruments instead of clearing all instruments simultaneously, as is the case with DCL.

GET (group execute trigger)

GET is a GPIB trigger that is used as an arm event to control operation.

SPE, SPD (serial polling)

Use the serial polling sequence to obtain the Model 2510 serial poll byte. The serial poll byte contains important information about internal functions. (See "Status structure" in Section 7.) Generally, the serial polling sequence is used by the controller to determine which of several instruments has requested service with the SRQ line. However, the serial polling sequence may be performed at any time to obtain the status byte from the Model 2510.

Programming syntax

The information in this section covers syntax for both common commands and SCPI commands. For information not covered here, see the IEEE- 488.2 and SCPI standards. See "Common commands" in Section 8 and "SCPI command reference" in Section 10 for details.

Command words

Program messages are made up of one or more command words.

Commands and command parameters

Common commands and SCPI commands may or may not use a parameter. The following are some examples:

*SAV <NRf>	Parameter (NRf) required
*RST	No parameter used
:SOURce:CURRent:LCONstants <n>	Parameter <n> required
:SYSTem:PRESet	No parameter used

NOTE *At least one space between the command word and the parameter is required.*

Brackets [ ] – Some command words are enclosed in brackets ([ ]). These brackets are used to denote an optional command word that does not need to be included in the program message. For example:

:INITiate[:IMMediate]

These brackets indicate that :IMMediate is implied (optional) and does not have to be used. Thus, the above command can be sent in one of two ways:

:INITiate
or
:INITiate:IMMediate

NOTE *The optional command is used without the brackets. When using optional command words in your program, do not include the brackets.*

Parameter types – The following are some of the more common parameter types:

<b> Boolean – Used to enable or disable an instrument operation. 0 or OFF disables the operation, and 1 or ON enables the operation. Example:

:SOUR:TEMP:PROT:STAT ON Enable temperature protection.

<name> Name parameter – Select a parameter name from a listed group.

<NRf> Numeric representation format – This parameter is a number that can be expressed as an integer (e.g., 8), a real number (e.g., 23.6), or an exponent (2.3E6).

<n> Numeric value – A numeric value parameter can consist of an NRf number or one of the following name parameters: DEFault, MINimum, MAXimum. When the DEFault parameter is used, the instrument is programmed to the *RST default value. When the MINimum parameter is used, the instrument is programmed to the lowest allowable value. When the MAXimum parameter is used, the instrument is programmed to the largest allowable value. Example:

:SOURce:CURRent 2 Sets current to 2A.

<numlist> Numlist – Specify one or more numbers for a list. Example:

:STATus:QUEue:ENABle (-110:-222)	Enable errors -110 through -222

<NDN> Non-decimal numeric – This parameter is used to send values in the binary, octal, or hexadecimal format. The prefix designates the format type:

#Bxx...x	#B specifies the binary format. xx...x is the binary number (using 0s and 1s).
#Qxx...x	#Q specifies the octal format. xx...x is the octal number (values 0 through 7).
#Hxx...x	#H specifies the hexadecimal format. xx...x is the hexadecimal number (values 0 through 9 and A through F).

Examples to send the decimal value 36 in the non-decimal formats:

*ESE #b100100	Binary format
*ESE #q44	Octal format
*ESE #h24	Hexadecimal format

Angle brackets < > – Angle brackets (< >) are used to denote a parameter type. Do not include the brackets in the program message.

Query commands

This type of command requests (queries) the presently programmed status. It is identified by the question mark (?) at the end of the fundamental form of the command. Most commands have a query form:

:SOURce:VOLTage?	Queries the output voltage.

Case sensitivity

Common commands and SCPI commands are not case sensitive. You can use upper or lower case and any case combination. Examples:

*RST	= *rst
:SYSTem:PRESet	= :system:preset

NOTE *Using all upper case will result in slightly faster command response times.*

Leading colon

The leading colon is not necessary, and not using the leading colon will slightly increase response times. Example:

:FORM:ELEM = FORM:ELEM

NOTE The "" for common commands is always required.*

Long-form and short-form versions

A SCPI command word can be sent in its long-form or short-form version. The command subsystem tables in Section 10 provide the long-form version. However, the short-form version is indicated by upper case characters. Examples:

:SYSTem:PRESet	long-form
:SYST:PRES	short-form
:SYSTem:PRES	long-form and short-form combination

Note that each command word must be in long-form or short-form, and not something in between. For example, :SYSTe:PRESe is illegal and will generate an error. The command will not be executed.

Short-form rules

Use the following rules to determine the short-form version of any SCPI command:

- If the length of the command word is four letters or less, no short form version exists. Example:

  :auto = :auto

These rules apply to command words that exceed four letters:

- If the fourth letter of the command word is a vowel (including "y"), delete it and all the letters after it. Example

  :immediate = :imm
- If the fourth letter of the command word is a consonant, retain it but drop all the letters after it. Example:

  :display = :disp
- If the command contains a question mark (?; query) or a non-optional number included in the command word, you must include it in the short-form version. Examples:

  :current? = :curr?
  :window2 = :wind2
  :window[1] = :wind
- Command words or characters that are enclosed in brackets ([ ]) are optional and need not be included in the program message.

NOTE For fastest response to commands, always use short forms.

Program messages

A program message is made up of one or more command words sent by the computer to the instrument. Each common command is a three letter acronym preceded by an asterisk (*). SCPI commands are categorized in the :STATus subsystem and are used to explain how command words are structured to formulate program messages.

:STATus	Path (Root)
:OPERation	Path
:ENABle <NRf>	Command and parameter
:ENABle?	Query command
:PRESet	Command

Single command messages

The above command structure has three levels. The first level is made up of the root command (:STATus) and serves as a path. The second level is made up of another path (:OPERation) and a command (:PRESet). The third path is made up of one command for the :OPERation path. The three commands in this structure can be executed by sending three separate program messages as follows:

```
:stat:oper:enab <NRf>
:stat:oper:enab?
:stat:pres
```

In each of the above program messages, the path pointer starts at the root command (:stat) and moves down the command levels until the command is executed.

Multiple command messages

You can send multiple command messages in the same program message as long as they are separated by semicolons (;). The following is an example showing two commands in one program message:

```
:stat:oper:enab <NRf>; :stat:oper:enab?
```

When the above is sent, the first command word is recognized as the root command (:stat). When the next colon is detected, the path pointer moves down to the next command level and executes the command. When the path pointer sees the colon after the semicolon (;), it resets back to the root level and starts over.

Commands that are on the same command level can be executed without having to retype the entire command path. Example:

```
:stat:oper:enab <NRf>; enab?
```

After the first command (:enab) is executed, the path pointer is at the third command level in the structure. Since :enab? is also on the third level, it can be typed in without repeating the entire path name. Notice that the leading colon for :enab? is not included in the program message. If a colon were included, the path pointer would reset to the root level and expect a root command. Since :enab? is not a root command, an error would occur.

Command path rules

- Each new program message must begin with the root command, unless it is optional (e.g., [:SENSe]). If the root is optional, simply treat a command word on the next level as the root. For fastest operation, do not send optional data.
- The colon (:) at the beginning of a program message is optional and need not be used. However, eliminating the first colon will result in fastest operation. Example:

  :stat:pres = stat:pres
- When the path pointer detects a colon (:) it moves down to the next command level. An exception is when the path pointer detects a semicolon (;), which is used to separate commands within the program message (see next rule).
- When the path pointer detects a colon (:) that immediately follows a semicolon (;), it resets back to the root level.
- The path pointer can only move down. It cannot be moved up a level. Executing a command at a higher level requires that you start over at the root command.

Using common and SCPI commands in the same message

Both common commands and SCPI commands can be used in the same message as long as they are separated by semicolons (;). A common command can be executed at any command level and will not affect the path pointer. Example:

:stat:oper:enab <NRf>; *ESE <NRf>

Program message terminator (PMT)

Each program message must be terminated with an LF (line feed), EOI (end or identify), or an LF+EOI. The bus will hang if your computer does not provide this termination. The following example shows how a multiple command program message must be terminated:

:sour:curr 1; :sour:volt5 <PMT>

Command execution rules

- Commands execute in the order that they are presented in the program message.
- An invalid command generates an error and, of course, is not executed.
- Valid commands that precede an invalid command in a multiple command program message are executed.
- Valid commands that follow an invalid command in a multiple command program message are ignored.

Response messages

A response message is the message sent by the instrument to the computer in response to a query command program message.

Sending a response message

After sending a query command, the response message is placed in the Output Queue. When the Model 2510 is then addressed to talk, the response message is sent from the Output Queue to the computer.

Multiple response messages

If you send more than one query command in the same program message (see "Multiple command messages"), the multiple response messages for all the queries are sent to the computer when the Model 2510 is addressed to talk. The responses are sent in the order the query commands were sent and are separated by semicolons (;). Items within the same query are separated by commas (,). The following example shows the response message for a program message that contains four single item query commands:

0; 1; 1; 0

Response message terminator (RMT)

Each response is terminated with an LF (line feed) and EOI (end or identify). The following example shows how a multiple response message is terminated:

0; 1; 1; 0 <RMT>

Message exchange protocol

Two rules summarize the message exchange protocol:

Rule 1. You must always tell the Model 2510 what to send to the computer.

The following two steps must always be performed to send information from the instrument to the computer:

1. Send the appropriate query command(s) in a program message.
2. Address the Model 2510 to talk.

Rule 2. The complete response message must be received by the computer before another program message can be sent to the Model 2510.

7
Status Structure

- **Overview** — Provides an operational overview of the status structure for the Model 2510.
- **Clearing registers and queues** — Covers the actions that clear (reset) registers and queues.
- **Programming and reading registers** — Explains how to program enable registers and read any register in the status structure.
- **Status byte and service request (SRQ)** — Explains how to program the Status Byte to generate service requests (SRQs). Shows how to use the serial poll sequence to detect SRQs.
- **Status register sets** — Provides bit identification and command information for the four status register sets: Standard Event Status, Operation Event Status, Measurement Event Status, and Questionable Event Status.
- **Queues** — Provides details and command information on the Output Queue and Error Queue.

Overview

The Model 2510 provides a series of status registers and queues allowing the operator to monitor and manipulate the various instrument events. The status structure is shown in Figure 7-1. The heart of the status structure is the Status Byte Register. This register can be read by the user's test program to determine if a service request (SRQ) has occurred, and what event caused it.

Status byte and SRQ

The Status Byte Register receives the summary bits of four status register sets and two queues. The register sets and queues monitor the various instrument events. When an enabled event occurs, it sets a summary bit in the Status Byte Register. When a summary bit of the Status Byte is set and its corresponding enable bit is set (as programmed by the user), the RQS/MSS bit will set to indicate that an SRQ has occurred.

Status register sets

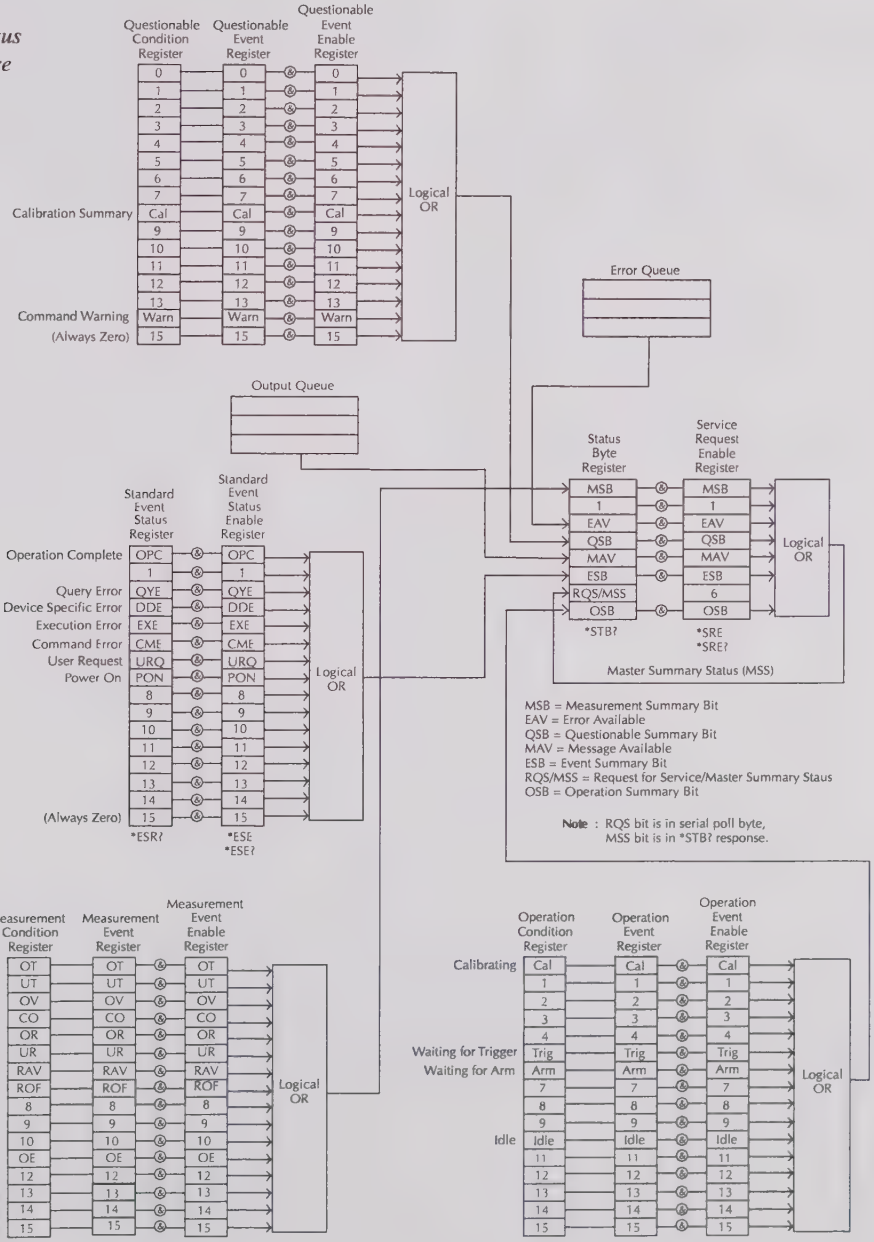
A typical status register set is made up of a condition register, an event register and an event enable register. A condition register is a read-only register that constantly updates to reflect the present operating conditions of the instrument.

When an event occurs, the appropriate event register bit sets to 1. The bit remains latched to 1 until the register is reset. When an event register bit is set and its corresponding enable bit is set (as programmed by the user), the output (summary) of the register will set to 1, which in turn sets the summary bit of the Status Byte Register.

Queues

The Model 2510 uses an Output Queue and an Error Queue. The response messages to query commands are placed in the Output Queue. As various programming errors and status messages occur, they are placed in the Error Queue. When a queue contains data, it sets the appropriate summary bit of the Status Byte Register.

Figure 7-1
Model 2510 status register structure



Clearing registers and queues

When the Model 2510 is turned on, the bits of all registers in the status structure are cleared (reset to 0), and the two queues are empty. Commands to reset the event and event enable registers, and the Error Queue are listed in Table 7-1. In addition to these commands, any enable register can be reset by sending the 0 parameter value with the individual command to program the register.

NOTE *SYSTem:PRESet and *RST have no effect on status structure registers and queues.*

Table 7-1
Common and SCPI commands to reset registers and clear queues

Commands	Description	Ref.
To Reset Registers:		
*CLS	Reset all bits of the following event registers to 0: Standard Event Register Operation Event Register Measurement Event Register Questionable Event Register	Note 1
:STATus:PRESet	Reset all bits of the following enable registers to 0: Operation Event Enable Register Measurement Event Enable Register Questionable Event Enable Register	Note 1
To Clear Error Queue:		
*CLS	Clear all messages from Error Queue	Note 2
:STATus:QUEue:CLEar	Clear messages from Error Queue	Note 3
:SYSTem:ERRor:CLEar	Clear messages from Error Queue	Note 3

Notes:
1. The Standard Event Enable Register is not reset by STATus:PRESet or *CLS. Send the 0 parameter value with *ESE to reset all bits of that enable register to 0 (see "Status byte and service request commands").
2. STATus:PRESet has no effect on the Error Queue.
3. Use either of the two clear commands to clear the Error Queue.

Programming and reading registers

Programming enable registers

The only registers that can be programmed by the user are the enable registers. All other registers in the status structure are read-only registers. The following explains how to ascertain the parameter values for the various commands used to program enable registers. The actual commands are covered later in this section (see Tables 7-3 and 7-6).

A command to program an event enable register is sent with a parameter value that determines the desired state (0 or 1) of each bit in the appropriate register. An enable register can be programmed using any of the following data formats for the parameter value; binary, decimal, hexadecimal, or octal.

The bit positions of the register (see Figure 7-2) indicate the binary parameter value. For example, if you wish to sets bits B4, B3, and B1, the binary value would be 11010 (where B4=1, B3=1, B1=1, and all other bits are 0). When you use one of the other formats, convert the binary number to its decimal, hexadecimal, or octal equivalent:

Binary 11010 = Decimal 26 = Hexadecimal 1A = Octal 32

Note that Figure 7-2 includes the decimal weight for each register bit. To set bits B4, B3, and B1, the decimal parameter value would be the sum of the decimal weights for those bits (16+8+2 = 26).

Figure 7-2
16-bit status register

A) Bits 0 through 7

Bit Position	B7	B6	B5	B4	B3	B2	B1	B0
Binary Value	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
Decimal Weights	128 (2^7)	64 (2^6)	32 (2^5)	16 (2^4)	8 (2^3)	4 (2^2)	2 (2^1)	1 (2^0)

B) Bits 8 through 15

Bit Position	B15	B14	B13	B12	B11	B10	B9	B8
Binary Value	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
Decimal Weights	32768 (2^{15})	16384 (2^{14})	8192 (2^{13})	4096 (2^{12})	2048 (2^{11})	1024 (2^{10})	512 (2^9)	256 (2^8)

The <NDN> (non-decimal numeric) parameter type is used to send non-decimal values. These values require a header (#B, #H or #Q) to identify the data format being sent. The letter in the header can be upper or lower case. The <NRf> (numeric representation format) parameter type is used to send decimal values, and does not use a header. The following examples show the proper parameter syntax for setting Bits B5, B3, and B2:

#b101100	Binary format (<NDN> parameter type)
#h2C	Hexadecimal format (<NDN> parameter type)
#q54	Octal format (<NDN> parameter type)
44	Decimal format (<NRf> parameter type)

Valid characters for the non-decimal parameter values are shown as follows:

<NDN> format	Valid characters
Binary	1s and 0s
Hexadecimal	0 through 9 and A through F
Octal	0 through 7

Reading registers

Any register in the status structure can be read by using the appropriate query (?) command. The following explains how to interpret the returned value (response message). The actual query commands are covered later in this section.

The response message will be a value that indicates which bits in the register are set. That value (if not already binary) will have to be converted to its binary equivalent. For example, for a binary value of 100101, bits B5, B2, and B0 are set.

The returned value can be in the binary, decimal, hexadecimal, or octal format. The FORMat:SREGister command is used to select the data format for the returned value (see Table 7-2).

For non-decimal formats, one of the following headers will accompany the returned value to indicate which format is selected:

#B = Header for binary values
#H = Header for hexadecimal values
#Q = Header for octal values

Table 7-2
Data format commands for reading status registers

Command	Description	Default
:FORMat:SREGister <name>	Select data format for reading status registers: <name> = ASCii — Decimal format HEXadecimal — Hexadecimal format OCTal — Octal format BINary — Binary format	ASCii

Status byte and service request (SRQ)

Service request is controlled by two 8-bit registers; the Status Byte Register and the Service Request Enable Register. Figure 7-3 shows the structure of these registers.

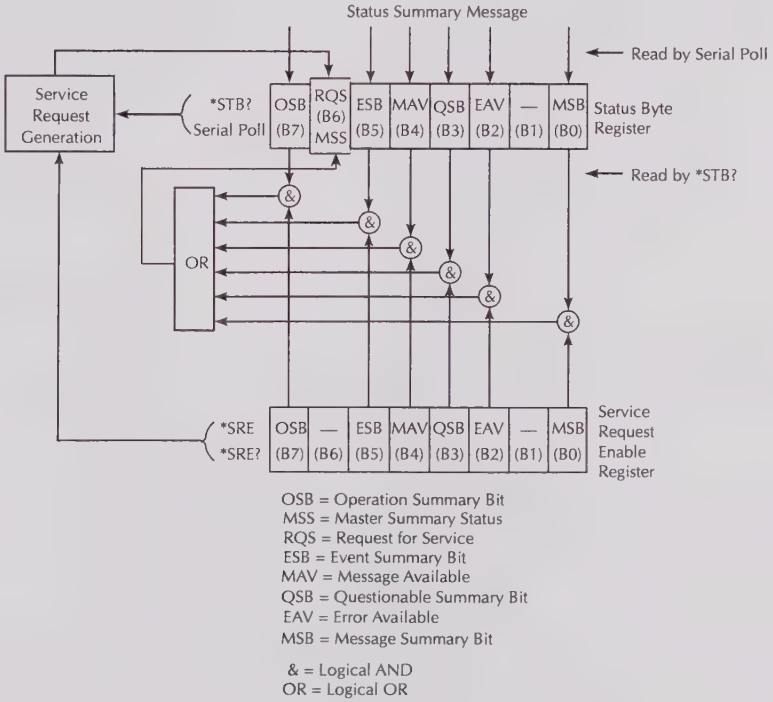


Figure 7-3
Status byte and service request (SRQ)

Status byte register

The summary messages from the status registers and queues are used to set or clear the appropriate bits (B0, B2, B3, B4, B5, and B7) of the Status Byte Register. These summary bits do not latch, and their states (0 or 1) are solely dependent on the summary messages (0 or 1). For example, if the Standard Event Register is read, its register will clear. As a result, its summary message will reset to 0, which in turn will reset the ESB bit in the Status Byte Register.

The bits of the Status Byte Register are described as follows:

- **Bit B0, Measurement Summary Bit (MSB)** — Set summary bit indicates that an enabled measurement event has occurred.
- **Bit B1** — Not used.
- **Bit B2, Error Available (EAV)** — Set summary bit indicates that an error or status message is present in the Error Queue.
- **Bit B3, Questionable Summary Bit (QSB)** — Set summary bit indicates that an enabled questionable event has occurred.
- **Bit B4, Message Available (MAV)** — Set summary bit indicates that a response message is present in the Output Queue.
- **Bit B5, Event Summary Bit (ESB)** — Set summary bit indicates that an enabled standard event has occurred.
- **Bit B6, Request Service (RQS)/Master Summary Status (MSS)** — Set bit indicates that an enabled summary bit of the Status Byte Register is set.
- **Bit B7, Operation Summary (OSB)** — Set summary bit indicates that an enabled operation event has occurred.

Depending on how it is used, Bit B6 of the Status Byte Register is either the Request for Service (RQS) bit or the Master Summary Status (MSS) bit:

- When using the serial poll sequence of the Model 2510 to obtain the status byte (a.k.a. serial poll byte), B6 is the RQS bit. See "Serial polling and SRQ" for details on using the serial poll sequence.
- When using the *STB? command (see Table 7-3) to read the status byte, B6 is the MSS bit.

Service request enable register

The generation of a service request is controlled by the Service Request Enable Register. This register is programmed by you and is used to enable or disable the setting of bit B6 (RQS/MSS) by the Status Summary Message bits (B0, B2, B3, B4, B5, and B7) of the Status Byte Register. As shown in Figure 7-3, the summary bits are logically ANDed (&) with the corresponding enable bits of the Service Request Enable Register. When a set (1) summary bit is ANDed with an enabled (1) bit of the enable register, the logic "1" output is applied to the input of the OR gate and, therefore, sets the MSS/RQS bit in the Status Byte Register.

The individual bits of the Service Request Enable Register can be set or cleared by using the *SRE common command. To read the Service Request Enable Register, use the *SRE? query command. The Service Request Enable Register clears when power is cycled or a parameter value of 0 is sent with the *SRE command (i.e. *SRE 0). The commands to program and read the SRQ Enable Register are listed in Table 7-3.

Serial polling and SRQ

Any enabled event summary bit that goes from 0 to 1 will set bit B6 and generate an SRQ (service request). In your test program, you can periodically read the Status Byte to check if an SRQ has occurred and what caused it. If an SRQ occurs, the program can, for example, branch to an appropriate subroutine that will service the request.

Typically, SRQs are managed by the serial poll sequence of the Model 2510. If an SRQ does not occur, bit B6 (RQS) of the Status Byte Register will remain cleared, and the program will simply proceed normally after the serial poll is performed. If an SRQ does occur, bit B6 of the Status Byte Register will set, and the program can branch to a service subroutine when the SRQ is detected by the serial poll.

The serial poll automatically resets RQS of the Status Byte Register. This allows subsequent serial polls to monitor bit B6 for an SRQ occurrence generated by other event types. After a serial poll, the same event can cause another SRQ, even if the event register that caused the first SRQ has not been cleared.

The serial poll does not clear MSS. The MSS bit stays set until all Status Byte summary bits are reset.

SPE, SPD (serial polling)

The SPE, SPD General Bus Command sequence is used to serial poll the Model 2510. Serial polling obtains the serial poll byte (status byte). Typically, serial polling is used by the controller to determine which of several instruments has requested service with the SRQ line.

Status byte and service request commands

The commands to program and read the Status Byte Register and Service Request Enable Register are listed in Table 7-3. For details on programming and reading registers, see "Programming enable registers and Reading registers."

NOTE *To reset the bits of the Service Request Enable Register to 0, use 0 as the parameter value for the *SRE command (i.e. *SRE 0).*

Table 7-3
Status byte and service request enable register commands

Command	Description	Default
*STB?	Read Status Byte Register	
*SRE <NDN> or <NRf>	Program the Service Request Enable Register: <NDN> = #Bxx...x Binary format (each x = 1 or 0) = #Hx Hexadecimal format (x = 0 to FF) = #Qx Octal format (x = 0 to 377) <NRf> = 0 to 255 Decimal format	(Note)
*SRE?	Read the Service Request Enable Register	

Note: *CLS and STATus:PRESet have no effect on the Service Request Enable Register.

Programming example - set MSS (B6) when error occurs

The first command of sequence in Table 7-4 enables EAV (error available). When an invalid command is sent (line 4), bits B2 (EAV) and B6 (MSS) of the Status Byte Register set to 1. The last command reads the Status Byte Register using the binary format (which directly indicates which bits are set). The command to select format (FORMat:SREGister) is documented in Table 7-2. To determine the exact nature of the error, you will have to read the Error Queue (see "Queues").

Table 7-4
Status byte programming example

Command	Description
*CLS	Clear Error Queue
*SRE 4	Enable EAV
FORM:SREG BIN	Select binary format
*XYZ	Generate error
*STB?	Read Status Byte Register

Status register sets

As shown in Figure 7-1, there are four status register sets in the status structure of the Model 2510; Standard Event Status, Operation Event Status, Measurement Event Status, and Questionable Event Status.

NOTE *See Appendix B for details on which register bits are set by specific status and error conditions.*

Register bit descriptions

Standard event register

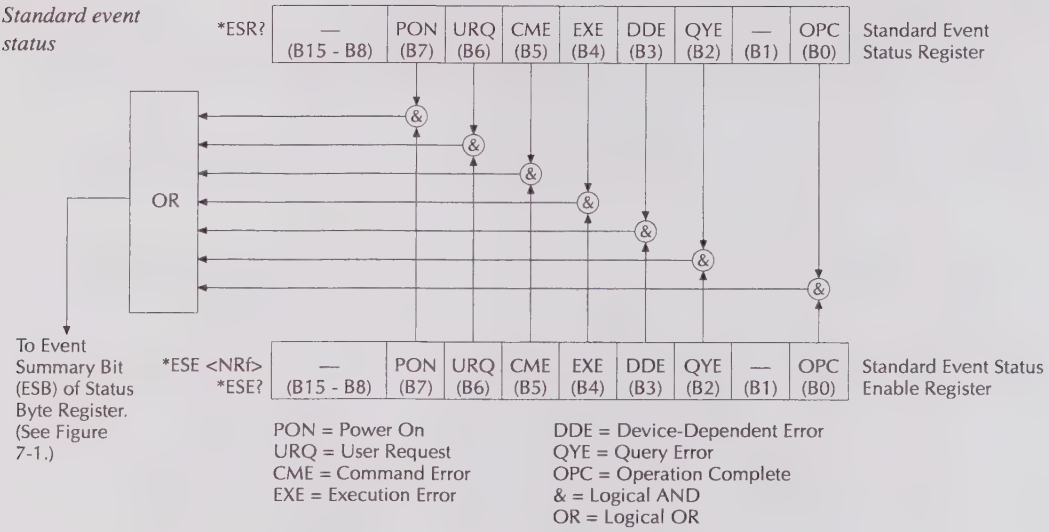
The used bits of the Standard Event Register (shown in Figure 7-4) are described as follows:

- **Bit B0, Operation Complete** — Set bit indicates that all pending selected device operations are completed and the Model 2510 is ready to accept new commands. This bit only sets in response to the *OPC? query command. See Section 8, "Common commands" for details on *OPC and *OPC?.
- **Bit B1** — Not used.
- **Bit B2, Query Error (QYE)** — Set bit indicates that you attempted to read data from an empty Output Queue.
- **Bit B3, Device-Dependent Error (DDE)** — Set bit indicates that an instrument operation did not execute properly due to some internal condition.
- **Bit B4, Execution Error (EXE)** — Set bit indicates that the Model 2510 detected an error while trying to execute a command.
- **Bit B5, Command Error (CME)** — Set bit indicates that a command error has occurred.

  Command errors include:

  - IEEE-488.2 syntax error — Model 2510 received a message that does not follow the defined syntax of the IEEE-488.2 standard.
  - Semantic error — Model 2510 received a command that was misspelled or received an optional IEEE-488.2 command that is not implemented.
  - The instrument received a Group Execute Trigger (GET) inside a program message.
- **Bit B6, User Request (URQ)** — Set bit indicates that the LOCAL key on the Model 2510 front panel was pressed.
- **Bit B7, Power ON (PON)** — Set bit indicates that the Model 2510 has been turned off and turned back on since the last time this register has been read.

Figure 7-4
Standard event status

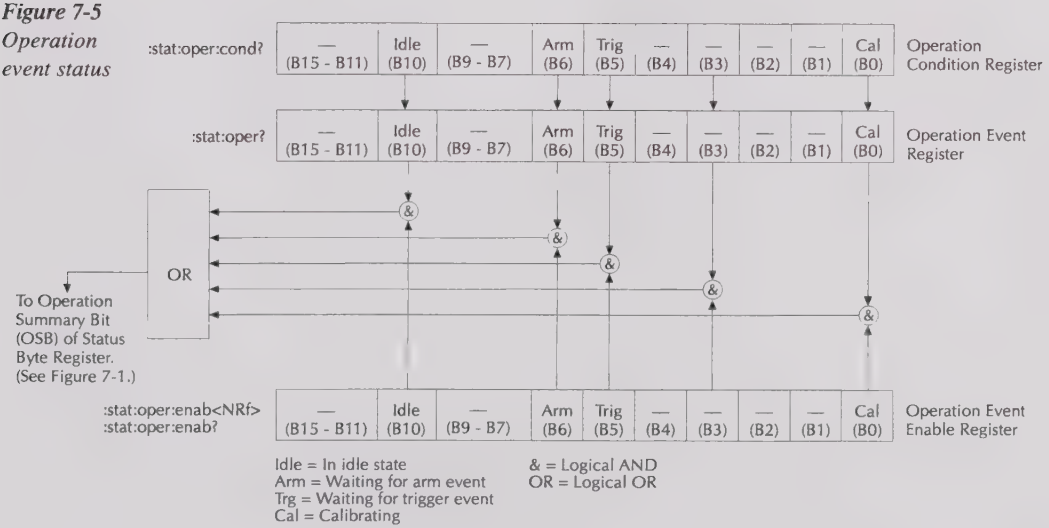


Operation event register

The used bits of the Operation Event Register (shown in Figure 7-5) are described as follows:

- **Bit B0, Calibrating (Cal)** — Set bit indicates that the Model 2510 is calibrating.
- **Bits B1 through B4** — Not used.
- **Bit B5, Waiting for Trigger Event (Trig)** — Set bit indicates that the Model 2510 is in the trigger layer waiting for a trigger event to occur.
- **Bit B6, Waiting for Arm Event (Arm)** — Set bit indicates that the Model 2510 is in the arm layer waiting for an arm event to occur.
- **Bits B7 through B9** — Not used (always 0).
- **Bit B10, Idle State (Idle)** — Set bit indicates the Model 2510 is in the idle state.
- **Bits B11 through B14** — Not used.
- **Bit B15** — Always zero.

Figure 7-5
Operation event status

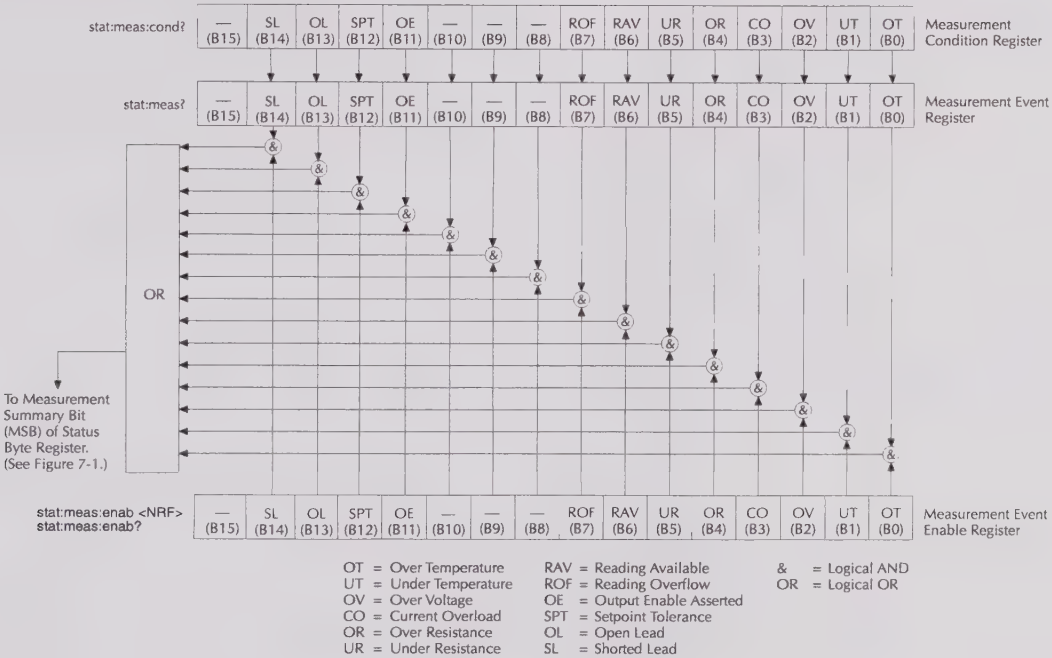


Measurement event register

The used bits of the Measurement Event Register (shown in Figure 7-6) are described as follows:

- **Bit B0, Over Temperature (OT)** — Set bit indicates an over temperature condition.
- **Bit B1, Under Temperature (UT)** — Set bit indicates an under temperature condition.
- **Bit B2, Over Voltage (OV)** — Set bit indicates an over voltage condition.
- **Bit B3, Current Overload (CO)** — Set bit indicates a current overload condition.
- **Bit B4, Over Resistance (OR)** — Set bit indicates an over resistance condition.
- **Bit B5, Under Resistance (UR)** — Set bit indicates an under resistance condition.
- **Bit B6, Reading Available (RAV)** — Set bit indicates that a reading was taken and processed.
- **Bit B7, Reading Overflow (ROF)** — Set bit indicates that the reading exceeds the selected measurement range of the Model 2510.
- **Bits B8 through B10** — Not used.
- **Bit B11, Output Enable (OE)** — Set bit indicates that the Digital I/O port output enable line is asserted.
- **Bit B12, Setpoint Tolerance (SPT)** — Set bit indicates that the setpoint is within the programmed tolerance for the required number of readings.
- **Bit B13, Open Lead (OL)** — Set bit indicates that the temperature sensor leads are open.
- **Bit B14, Shorted Lead (SL)** — Set bit indicates that the temperature sensor leads are shorted.
- **Bit B15** — Always zero.

Figure 7-6
Measurement event status


stat:meas:cond?
— (B15) SL (B14) OL (B13) SPT (B12) OE (B11) — (B10) — (B9) — (B8) ROF (B7) RAV (B6) UR (B5) OR (B4) CO (B3) OV (B2) UT (B1) OT (B0)
Measurement Condition Register
stat:meas?
— (B15) SL (B14) OL (B13) SPT (B12) OE (B11) — (B10) — (B9) — (B8) ROF (B7) RAV (B6) UR (B5) OR (B4) CO (B3) OV (B2) UT (B1) OT (B0)
Measurement Event Register
OR
To Measurement Summary Bit (MSB) of Status Byte Register. (See Figure 7-1.)
stat:meas:enab <NRF>
stat:meas:enab?
— (B15) SL (B14) OL (B13) SPT (B12) OE (B11) — (B10) — (B9) — (B8) ROF (B7) RAV (B6) UR (B5) OR (B4) CO (B3) OV (B2) UT (B1) OT (B0)
Measurement Event Enable Register
OT = Over Temperature
UT = Under Temperature
OV = Over Voltage
CO = Current Overload
OR = Over Resistance
UR = Under Resistance
RAV = Reading Available
ROF = Reading Overflow
OE = Output Enable Asserted
SPT = Setpoint Tolerance
OL = Open Lead
SL = Shorted Lead
& = Logical AND
OR = Logical OR

Questionable event register

The used bits of the Questionable Event Register (shown in Figure 7-7) are described as follows:

- **Bits B0 through B7** — Not used.
- **Bit B8, Calibration Summary (Cal)** — Set bit indicates that an invalid calibration constant was detected during the power-up sequence. This error will clear after successful calibration of the instrument.
- **Bits B9 through B13** — Not used.
- **Bit B14, Command Warning (Warn)** — Set bit indicates that a Signal Oriented Measurement Command parameter has been ignored.
- **Bit B15** — Always zero.

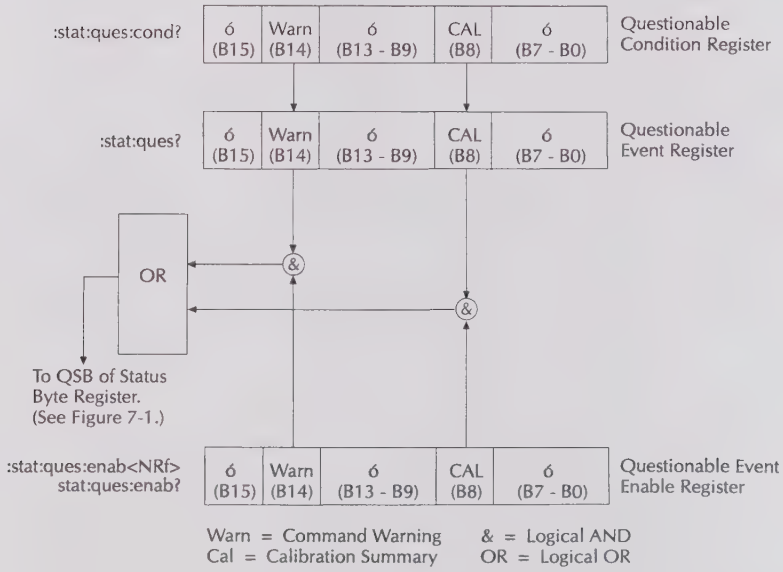


Figure 7-7 Questionable event status

Condition registers

As Figure 7-1 shows, each status register set (except the Standard Event Register set) has a condition register. A condition register is a real-time, read-only register that constantly updates to reflect the present operating conditions of the instrument. For example, while the Model 2510 is in the idle state, bit B10 (Idle) of the Operation Condition Register will be set. When the instrument is taken out of idle, bit B10 clears.

The commands to read the condition registers are listed in Table 7-5. For details on reading registers, see "Reading registers."

Table 7-5
Condition register commands

Command	Description
:STATus:OPERation:CONDition?	Read Operation Condition Register.
:STATus:MEASurement:CONDition?	Read Measurement Condition Register.
:STATus:QUEStionable:CONDition?	Read Questionable Condition Register.

Event registers

As Figure 7-1 shows, each status register set has an event register. When an event occurs, the appropriate event register bit sets to 1. The bit remains latched to 1 until the register is reset. Reading an event register clears the bits of that register. *CLS resets all four event registers.

The commands to read the event registers are listed in Table 7-6. For details on reading registers, see "Reading registers."

Table 7-6
Event register commands

Command	Description	Default
*ESR?	Read Standard Event Status Register.	(Note)
:STATus:OPERation:[:EVENt]?	Read Operation Event Register.	
:STATus:MEASurement:[:EVENt]?	Read Measurement Event Register.	
:STATus:QUEStionable:[:EVENt]?	Read Questionable Event Register.	

Note: Power-up and *CLS resets all bits of all event registers to 0. STATus:PRESet has no effect.

Event enable registers

As Figure 7-1 shows, each status register set has an enable register. Each event register bit is logically ANDed (&) to a corresponding enable bit of an enable register. Therefore, when an event bit is set and the corresponding enable bit is set (as programmed by the user), the output (summary) of the register will set to 1, which in turn sets the summary bit of the Status Byte Register.

The commands to program and read the event enable registers are listed in Table 7-7. For details on programming and reading registers, see "Programming enable registers and Reading registers."

NOTE *The bits of any enable register can be reset to 0 by sending the 0 parameter value with the appropriate enable command (i.e. STATus:OPERation:ENABle 0). *CLS has no effect on event enable registers.*

Table 7-7
Event enable registers commands

Command	Description	Default
*ESE <NDN> or <NRf>	Program Standard Event Enable Register. (See "Parameters".)	(Note)
*ESE?	Read Standard Event Enable Register.	
STATus	STATus Subsystem:	
:OPERation	Operation Event Enable Register:	
:ENABle <NDN> or <NRf>	Program enable register. (See "Parameters".)	
:ENABle?	Read enable register.	
:MEASurement	Measurement Event Enable Register:	
:ENABle <NDN> or <NRf>	Program enable register. (See "Parameters".)	
:ENABle?	Read enable register.	
:QUEStionable	Questionable Event Enable Register:	
:ENABle <NDN> or <NRf>	Program enable register. (See "Parameters".)	
:ENABle?	Read Measurement Event Enable Register.	
Parameters:		
<NDN> = #Bxx...x	Binary format (each x = 1 or 0)	
= #Hx	Hexadecimal format (x = 0 to FFFF)	
= #Qx	Octal format (x = 0 to 177777)	
<NRf> = 0 to 65535	Decimal format	

Note: Power-up and STATus:PRESet reset all bits of all enable registers to 0. *CLS has no effect.

Programming example - program and read register set

The command sequence in Table 7-8 programs and reads the measurement register set. Registers are read using the binary format (which directly indicates which bits are set). The command to select format (FORMat:SREGister) is documented in Table 7-2.

Table 7-8
Program and read register programming example

Command	Description
FORM:SREG BIN	Select binary format to read registers.
STAT:MEAS:ENAB 512	Enable BFL (buffer full).
STAT:MEAS:COND?	Read Measurement Condition Register.
STAT:MEAS?	Read Measurement Event Register.

Queues

The Model 2510 uses two queues, which are first-in, first-out (FIFO) registers:

- Output Queue — Used to hold reading and response messages.
- Error Queue — Used to hold error and status messages. (See Appendix B.)

The Model 2510 status model (Figure 7-1) shows how the two queues are structured with the other registers.

Output queue

The output queue holds data that pertains to the normal operation of the instrument. For example, when a query command is sent, the response message is placed in the Output Queue.

When data is placed in the Output Queue, the Message Available (MAV) bit in the Status Byte Register sets. A data message is cleared from the Output Queue when it is read. The Output Queue is considered cleared when it is empty. An empty Output Queue clears the MAV bit in the Status Byte Register.

A message is read from the Output Queue by addressing the Model 2510 to talk after the appropriate query is sent.

Error queue

The Error Queue holds error and status messages. When an error or status event occurs, a message that defines the error/status is placed in the Error Queue.

When a message is placed in the Error Queue, the Error Available (EAV) bit in the Status Byte Register is set. An error/status message is cleared from the Error Queue when it is read. The Error Queue is considered cleared when it is empty. An empty Error Queue clears the EAV bit in the Status Byte Register.

The Error Queue holds up to 10 error/status messages. The commands to read the Error Queue are listed in Table 7-9. When you read a single message in the Error Queue, the "oldest" message is read and then removed from the queue. If the queue becomes full, the message "350, 'Queue Overflow'" will occupy the last memory location. On power-up, the Error Queue is empty. When empty, the message "0, No Error" is placed in the queue.

Messages in the Error Queue are preceded by a code number. Negative (-) numbers are used for SCPI-defined messages, and positive (+) numbers are used for Keithley-defined messages. As shown in Table 7-9, there are commands to read the entire message (code and message) or the code only.

On power-up, all error messages are enabled and will go into the Error Queue as they occur. Status messages are not enabled and will not go into the queue. As listed in Table 7-9, there are commands to enable and/or disable messages. For these commands, the <list> parameter is used to specify which messages to enable or disable. The messages are specified by their codes. The following examples show various forms for using the <list> parameter.

<list>	= (-110)	Single message
	= (-110:-222)	Range of messages (-110 through -222)
	= (-110:-222, -220)	Range entry and single entry (separated by a comma)

When you enable messages, messages not specified in the list are disabled. When you disable messages, each listed message is removed from the enabled list.

NOTE *To prevent all messages from entering the Error Queue, send the enable command along with the null list parameter as follows: STATus:QUEue:ENABle ().*

Table 7-9
Error queue commands

Command	Description	Default
STATus	STATus Subsystem:	
:QUEue	Read Error Queue:	(Note 1)
[:NEXT]?	Read and clear oldest error/status (code and message).	
:ENABle <list>	Specify error and status messages for Error Queue.	(Note 2)
:ENABle?	Read the enabled messages.	
:DISable <list>	Specify messages not to be placed in queue.	(Note 2)
:DISable?	Read the disabled messages.	
:CLEar	Clear messages from Error Queue.	
SYSTem	SYSTem Subsystem:	
:ERRor	Read Error Queue:	(Note 1)
[:NEXT]?	Read and clear oldest error/status (code and message).	
:ALL?	Read and clear all errors/status (code and message).	
:COUNt?	Read the number of messages in queue.	
:CODE	Code numbers only:	
[:NEXT]?	Read and clear oldest error/status (code only).	
:ALL?	Read and clear all errors/status (codes only).	
:CLEar	Clear messages from Error Queue.	

Notes:
1. Power-up and *CLS empties the Error Queue. STATus:PRESet has no effect.
2. Power-up enables error messages and disables status messages. *CLS and STATus:PRESet have no effect.

Programming example - read error queue

The following command reads the error queue:

```
STAT:QUE?
```

8
Common Commands

- **Command summary** — Lists the IEEE-488.2 common commands used by the Model 2510.
- **Command reference** — Provides a detailed reference for all common commands except for those associated with the status structure, which are discussed in Section 7.

Command summary

Common commands (summarized in Table 8-1) are device commands that are common to all devices on the bus. These commands are designated and defined by the IEEE-488.2 standard. Most of these commands are described in detail in this section.

NOTE *The following common commands associated with the status structure are covered in "Status structure" in Section 7: *CLS, *ESE, *ESE?, *ESR?, *SRE, *SRE?, and *STB?.*

Table 8-1
IEEE-488.2 common commands and queries

Mnemonic	Name	Description
*CLS	Clear status	Clears all event registers and Error Queue.[1]
*ESE <NRf>	Event enable command	Program the Standard Event Enable Register.[1]
*ESE?	Event enable query	Read the Standard Event Enable Register.[1]
*ESR?	Event status register query	Read and clear the Standard Event Enable Register.[1]
*IDN?	Identification query	Returns the manufacturer, model number, serial number, and firmware revision levels of the unit.
*OPC	Operation complete command	Set the Operation Complete bit in the Standard Event Register after all pending commands have been executed.
*OPC?	Operation complete query	Places an ASCII "1" into the Output Queue when all pending selected device operations have been completed.
*OPT?	Option query	Returns summary of installed options.
*RCL <NRf>	Recall command	Returns the Model 2510 to the user-saved setup.
*RST	Reset command	Returns the Model 2510 to the *RST default conditions.
*SAV <NRf>	Save command	Saves the present setup as the user-saved setup.
*SRE <NRf>	Service request enable command	Programs the Service Request Enable Register.[1]
*SRE?	Service request enable query	Reads the Service Request Enable Register.[1]
*STB?	Status byte query	Reads the Status Byte Register.[1]
*TRG	Trigger command	Sends a bus trigger to the Model 2510.
*TST?	Self-test query	Performs a checksum test on ROM and returns the result.
*WAI	Wait-to-continue command	Wait until all previous commands are executed.

[1]Status commands are covered in "Status structure" in Section 7.

Command reference

*IDN? — identification query

Reads identification code

The identification code includes the manufacturer, model number, serial number, and firmware revision levels and is sent in the following format:

KEITHLEY INSTRUMENTS INC., MODEL 2510, xxxxxxx, yyyyy/zzzzz /b

Where: 2510 is the model number.

xxxxxxx is the serial number.

yyyyy/zzzzz is the firmware revision levels of the digital board ROM and display board ROM, including date and time of build.

b is the board revision level.

*OPC — operation complete

Sets OPC bit

*OPC? — operation complete query

Places a "1" in output queue

When *OPC is sent, the OPC bit in the Standard Event Register will set after all pending command operations are complete. When *OPC? is sent, an ASCII "1" is placed in the Output Queue after all pending command operations are complete.

Typically, either one of these commands is sent after the INITiate command. The INITiate command is used to take the instrument out of idle in order to perform measurements. While operating within the trigger model layers, all sent commands (except DCL, SDC, IFC, SYSTem:PRESet, *RST, *RCL, *TRG, GET, and ABORt) will not execute.

After all programmed operations are completed, the instrument returns to the idle state at which time all pending commands (including *OPC and/or *OPC?) are executed. After the last pending command is executed, the OPC bit and/or an ASCII "1" is placed in the Output Queue.

When *OPC is sent, the OPC bit in the Standard Event Register will set after all pending command operations are complete. When *OPC? is sent, an ASCII "1" is placed in the Output Queue after all pending command operations are complete.

*OPT? — option query

Return list of installed options

The *OPT query returns a list of any installed options in ASCII format.

*SAV <NRf> — save

Save present setup in memory

*RCL <NRf> — recall

Return to setup stored in memory

Parameters:
0 = Memory location 0
1 = Memory location 1
2 = Memory location 2
3 = Memory location 3
4 = Memory location 4

Use the *SAV command to save the present instrument setup configuration in non-volatile memory for later recall. Any control affected by *RST can be saved by the *SAV command. The *RCL command is used to restore the instrument to the saved setup configuration. Five setup configurations can be saved and recalled.

The Model 2510 ships from the factory with SYSTem:PRESet defaults loaded into the available setup memory. If a recall error occurs, the setup memory defaults to the SYSTem:PRESet values. See Tables 10-1 through 10-10 in Section 10 for :SYSTem:PRESet and *RST default conditions.

*RST — reset

Return Model 2510 to GPIB defaults

When the *RST command is sent, the Model 2510 performs the following operations:

- Returns the Model 2510 to the GPIB default conditions (see Section 10, "SCPI command reference").
- Cancels all pending commands.
- Cancels response to any previously received *OPC and *OPC? commands.

*TRG — trigger

Send bus trigger to Model 2510

Use the *TRG command to issue a GPIB trigger to the Model 2510. It has the same effect as a group execute trigger (GET).

*TST? — self-test query

Run self test and read result

Use this query command to perform a checksum test on ROM. The command places the coded result (0 or 1) in the Output Queue. When the Model 2510 is addressed to talk, the coded result is sent from the Output Queue to the computer.

A returned value of zero (0) indicates that the test passed, and a value of one (1) indicates that the test failed.

*WAI — wait-to-continue

Wait until previous commands are completed

Effectively, the *WAI command is a No-Op (no operation) for the Model 2510 and thus, does not need to be used.

Two types of device commands exist:

- Sequential commands — A command whose operations are allowed to finish before the next command is executed.
- Overlapped commands — A command that allows the execution of subsequent commands while device operations of the Overlapped command are still in progress.

The *WAI command is used to suspend the execution of subsequent commands until the device operations of all previous Overlapped commands are finished. The *WAI command is not needed for Sequential commands.

9
SCPI Signal-Oriented Measurement Commands

- **Command summary** — Summarizes those commands used to configure and acquire readings.
- **Acquiring readings** — Describes commands to acquire post-processed readings, both trigger and acquire readings, and to perform a single measurement.
- **Command examples** — Gives several examples using signal-oriented commands.

Command summary

The signal-oriented measurement commands are used to acquire readings such as thermoelectric cooler DC current, power, and resistance. You can use these high-level instructions to trigger and acquire readings directly without having to send any other commands.

Signal-oriented commands are summarized in Table 9-1.

NOTE *The :FORMat:ELEMents command determines the reading elements returned by MEAS?, READ?, and FETCH?. (See Section 10, ":FORMat subsystem.")*

Table 9-1
Signal-oriented measurement command summary

Command	Description
:MEASure	
:CURRent	
[DC]?	Query DC current measurement of thermoelectric cooler.
:POWer?	Query power measurement of thermoelectric cooler.
:RESistance	
[:DC]?	Query DC ohms measurement of thermoelectric cooler.
:AC?	Query AC ohms measurement of thermoelectric cooler.
:TEMPerature?	Query temperature measurement.
:TSENSor?	Query temperature sensor (very useful in SOUR:FUNC RES mode).
:VOLTage	
[:DC]?	Query DC voltage measurement of thermoelectric cooler.
:MEASure?	Performs an :INITiate and a :FETCh?
:READ?	Identical to :MEASure?.
:FETCh?	Requests latest readings from unit.

Acquiring readings

:FETCh?

Description This query command requests the latest post-processed readings stored in the sample buffer. After sending this command and addressing the Model 2510 to talk, the readings are sent to the computer. This command does not affect the instrument setup.

This command does not trigger control-measure operations; it simply requests the last available readings. Note that this command can repeatedly return the same readings. Until there are new readings, this command continues to return the old readings. The readings that are acquired depend

on which data elements are selected (Section 10, "FORMat subsystem, Data elements"), and what the instrument is presently programmed to control and measure. Measure readings take priority over source readings, and functions not sourced or measured are assigned the NAN (not a number) value of +9.91e37.

NOTE *The :FETCh? command is automatically asserted when the :READ? or :MEASure? command is sent.*

:MEASure[:<function>]?

Parameters

<function> =	CURRent[:DC]	Measure Peltier DC current
	POWer	Measure Peltier power
	RESistance[:DC]	Measure Peltier DC resistance
	RESistance:AC	Measure Peltier AC resistance
	TEMPerature	Measure Peltier temperature
	TSENsor	Measure temperature sensor
	VOLTage[:DC]	Measure Peltier DC voltage

Description

This command combines other signal-oriented measurement commands to perform a "one-shot" measurement and acquire the reading. Note that if a function is not specified (:MEASure?), the measurement will be done on the function that is presently selected. When this command is sent, the following commands execute in the order that they are presented:

- :INITiate
- :FETCh?

The :INITiate command starts operation by taking the instrument out of idle. After all control-measure operations are completed, the Model 2510 goes back into idle at which time the :FETCh? command is executed. The readings are sent to the computer and displayed when the unit is addressed to talk.

NOTE *MEAS:RES:AC? will turn the output OFF and leave it OFF.*

:READ?

Description This command is used to trigger and acquire readings and is identical to :MEASure?. When this command is sent, the following commands execute in the order that they are presented:

- :INITiate
- :FETCh?

The :INITiate command starts operation by taking the instrument out of idle. After all control-measure operations are completed, the Model 2510 goes back into idle at which time the :FETCh? command is executed. The readings are sent to the computer and displayed when the unit is addressed to talk.

:INITiate[:IMMediate]

Description This command is used to trigger readings, but it does not request readings from the unit. The :INITiate command starts operation by taking the instrument out of idle. After all control-measure operations are completed, the Model 2510 goes back into idle. There is an implied INIT with :READ? and :MEAS?, but the Model 2510 does not return to idle before the next command is executed since the instrument does not support overlapped :INIT. If a READ? or MEAS? command is sent while the Model 2510 is taking readings, no INIT ignored error will be generated.

Command examples

`:MEAS:TEMP?`	Query temperature.
`:MEAS:POW?`	Query thermoelectric cooler power.
`:MEAS:RES:AC?`	Query cooler AC resistance.

10
SCPI Command Reference

- **Reference tables** — Summarizes each SCPI command subsystem.
- **SCPI subsystems** — Provides detailed information on all commands in each SCPI subsystem.

Reference tables

The SCPI commands that control the Model 2510 are summarized in the table below, which lists the various subsystems and reference tables.

Subsystem	Description	Reference Table
:DISPlay	Control/read front panel display attributes.	10-1
:FORMat	Specify remote data formats.	10-2
:OUTPut	Turn source output on/off, control output enable line.	10-3
:SENSe	Control measurement function operating modes.	10-4
:SOURce[1]	Control source function operating modes.	10-5
:SOURce2	Set Digital I/O port line logic levels.	10-6
:STATus	Control and read status registers.	10-7
:SYSTem	Control system functions such as line frequency.	10-8
Trigger	Control instrument triggering.	10-9
:UNIT	Set temperature units.	10-10

General notes:

- Brackets ([ ]) are used to denote optional character sets. These optional characters need not be included in the program message. Also, do not include brackets in program messages.
- Angle brackets (< >) are used to indicate parameter type. Do not use angle brackets in the program message.
- The Boolean parameter (<b>) is used to enable or disable an instrument operation. ON or 1 enables the operation, and OFF or 0 disables the operation.
- Upper-case characters indicate the short-form version for each command word.
- **Default parameter:** Listed parameters are both *RST and :SYSTem:PRESet defaults unless otherwise noted.
- **SCPI:** A check mark (✓) indicates that the command and its parameters are SCPI confirmed. An unmarked command indicates that it is a SCPI command but does not conform to the SCPI standard of set commands, and it is not a command recognized by the SCPI consortium. SCPI commands that use one or more non-SCPI parameters are explained by notes.
- See Section 6, "Programming syntax" for more information.

Table 10-1
:DISPlay subsystem commands

Command	Description	Default Parameter	SCPI
:DISPlay			
[:WINDow[1]]			✓
:ATTRibutes?	Returns a 20 character string representing the attributes of the characters on the top line. Blinking characters are 1 and others are 0.		
:TEXT			✓
[:DATA] <string>	Sets a 20 character string to be displayed on the top line.	Note	✓
[:DATA]?	Returns the programmed string for the top line.		✓
:STATe <b>	Enables (1/ON) or disables (0/OFF) the display of the user defined text on the top line of the display.	Note	✓
:STATe?	Returns the state of the user defined message.		✓
:DATA?	Returns the 20 characters that are displayed on the top line.		
:WINDow2			
:ATTRibutes?	Returns a 32 character string representing the attributes of the characters on the bottom line. Blinking characters are 1 and others are 0.		
:TEXT			✓
[:DATA] <string>	Sets a 32 character string to be displayed on the bottom line.	Note	✓
[:DATA]?	Returns the programmed string for the bottom line.		✓
:STATe <b>	Enables (1/ON) or disables (0/OFF) the display of the user defined text on the bottom line of the display.	Note	✓
:STATe?	Returns the state of the user defined message.		✓
:DATA?	Returns the 32 characters that are displayed on the bottom line.		✓

Note: RST or :SYSTem:PRESet have no effect on display data or state.

Table 10-2
:FORMat subsystem commands

Command	Description	Default Parameter	SCPI
:FORMat			
:BORDer <name>	Specify byte order when transferring binary data. <name> =NORMal | SWAPped	Note	✓
:BORDer?	Query byte order.		✓
[:DATA] <type>[,<length>]	Specify data format used when transferring numeric data. Valid formats are: ASCii, REAL, 32 and SREal. ASCii is the only permitted input format regardless of this setting.	ASCii	✓
[:DATA]?	Query data format.		✓
:ELEMents <item list>	Specify desired data item: TEMPerature, VOLTage, CURRent, POWer, RESistance, TSENsor, TIME, and STATus.	TEMP, VOLT, CURR, POW, RES, TSEN, TIME, STATus	
:SREGister <type>	Type = ASCii, HEXadecimal, OCTal, or BINary. Selects the output response format of the Status Model registers.	ASCii	✓
:SREGister?	Query output response format.		
:SOURce2 <type>	Type = ASCii, HEXadecimal, OCTal, or BINary. Selects output response format of Digital I/O settings.	ASCii	
:SOURce2?	Query Digital I/O output response format.		

Note: *RST default: NORMal
:SYSTem:PRESet default: SWAPped.

Table 10-3
:OUTPut subsystem commands

Command	Description	Default Parameter	SCPI
:OUTPut[1]			✓
:ENABle	Path to control Digital I/O output enable line.		
[:STATe] <b>	Activate (ON/1) or inactivate (OFF/0) enable line.	OFF	
[:STATe]?	Query state of output enable line.		
:TRIPped?	Output enable line tripped? (1 = yes; 0 = no.)		
[:STATe]	Enable (ON/1) or disable (OFF/0) output.	OFF	✓
[:STATe]?	Query output state.		✓

Table 10-4
:SENSe[1] subsystem commands

Command	Description	Default Parameter	SCPI
[:SENSe[1]]			✓
:CURRent[:DC]	Path to configure DC current.		✓
:PROTection	Set current protection.		✓
[:LEVel] <n>	Set protection current level. <n> = 1.0 to 5.25 (A) | MIN | MAX | DEF.	2.0	✓
[:LEVel]?	Query protection current level.		✓
:TRIPped?	Query if protection level exceeded. (1 = tripped.)		✓
:RESistance	Path to configure resistance.		✓
:CURRent <n>	Set RTD or thermistor sensor current. <n> = 3e-6 to 1e-3 (3.3e-6, 1e-5, 3.33e-5, 1e-4, 8.333e-4, 2.5e-3) | MIN | MAX | DEF.	2.5e-3 (100Ω range, RTD) 8.333e-4 (1kΩ range) 1e-4 (10kΩ range) 3.33e-5 (100kΩ range)	
:AUTO <b>	Enable/disable default sensor current (ON | OFF).	ON	
:AUTO?	Query default sensor current state.		
:CURRent?	Query sensor current.		
:TRANsducer <name>	Select sensor type. <name> = THERmistor | RTD	THERmistor	
:TRANsducer?	Query transducer type.		
:THERmistor			
RANGe <n>	Set thermistor resistance range. <n> = 0 to 3e5 (100, 1e3, 1e4, 1e5) | MIN | MAX | DEF.	1e4	
RANGe?	Query thermistor resistance range.		
:TEMPerature	Path to configure temperature.		
:CURRent <n>	Set RTD or thermistor sensor current. <n> = 3e-6 to 1e-3 (3.3e-6, 1e-5, 3.33e-5, 1e-4, 8.333e-4, 2.5e-3) | MIN | MAX | DEF	2.5e-3 (100Ω range, RTD) 8.333e-4 (1kΩ range) 1e-4 (10kΩ range) 3.33e-5 (100kΩ range)	
:AUTO <b>	Enable/disable auto sensor current (ON | OFF).	ON	
:AUTO?	Query auto default sensor current state.		
:CURRent?	Query sensor current.		
:TRANsducer <name>	Select transducer (sensor) type. <name> = THERmistor | RTD | VSS | ISS	THERmistor	
:TRANsducer?	Query transducer type.		
:RTD	Program RTD type and coefficients.		
:TYPE <name>	Select RTD sensor type. <name> = PT385, PT3916, PT100, D100, F100, or USER.	PT385	

Table 10-4 (cont.)
:SENSe[1] subsystem commands

Command	Description	Default Parameter	SCPI
[:SENSe[1]]			✓
:TEMPerature			
:RTD			
:ALPHa <n>	Program alpha coefficient for USER type. <n> = 0 to 0.01 | MIN | MAX | DEF	0.00385	
:ALPHa?	Query alpha coefficient.		
:BETA <n>	Program beta coefficient for USER type. <n> = 0 to 1 | MIN | MAX | DEF	0.111	
:BETA?	Query beta coefficient.		
:DELTa <n>	Program delta coefficient for USER type. <n> = 0 to 5 | MIN | MAX | DEF	1.507	
:DELTa?	Query delta coefficient.		
:THERmistor	Program thermistor parameters.		
:RANGe <n>	Set thermistor resistance range. <n> = 0 to 3e5 (100, 1e3, 1e4, 1e5) | MIN | MAX | DEF	1e4	
:RANGe?	Query thermistor resistance range.		
:A <n>	Set A parameter value. <n> = -10 to +10 | MIN | MAX | DEF	1.12396e-3	
:A?	Query A parameter value.		
:B <n>	Set B parameter value. <n> = -10 to +10 | MIN | MAX | DEF	2.30093e-4	
:B?	Query B parameter value.		
:C <n>	Set C parameter value. <n> = -10 to +10 | MIN | MAX | DEF	7.06351e-8	
:C?	Query C parameter value.		
:VSS	Set voltage type solid-state sensor parameters.		
[:GAIN] <n>	Program sensor gain (scale factor). <n> = 0 to 9.999e-2 | MIN | MAX | DEF	1.0e-2	
[:GAIN]?	Query sensor gain.		
:OFFSet <n>	Program sensor offset. <n> = 0 to 999.999 | MIN | MAX | DEF	0.0	
:OFFSet?	Query sensor offset.		
:ISS	Set current type solid-state sensor parameters.		
[:GAIN] <n>	Program sensor gain (scale factor). <n> = 0 to 9.999e-4 | MIN | MAX | DEF	1.0e-6	
[:GAIN]?	Query sensor gain.		
:OFFSet <n>	Program sensor offset. <n> = 0 to 999.999 | MIN | MAX | DEF	0.0	
:OFFSet?	Query sensor offset.		

Table 10-5
:SOURce[1] subsystem commands

Command	Description	Default Parameter	SCPI
:SOURce[1]	Path to SOURce[1] commands.		✓
:CLEar			✓
[:IMMediate]	Clear source immediately.		✓
:CURRent	Program current source.		✓
:LCONstants	Program current PID loop constants.		
[:GAIN] <n>	Set current loop gain. <n> = 0 to 999.999 | MIN | MAX | DEF	10.0	
[:GAIN]?	Query current loop gain.		
:DERivative <n>	Set current derivative constant. <n> = 0 to 100 | MIN | MAX | DEF	0.0	
:DERivative?	Query current derivative constant.		
:INTegral <n>	Set current integral constant. <n> = 0 to 100 | MIN | MAX | DEF	20.0	
:INTegral?	Query current integral constant.		
[:LEVel]			✓
[:IMMediate]			✓
[:AMPLitude] <n>	Set current amplitude. <n> = -5 to +5 (A) | MIN | MAX | DEF	0.0	✓
[:AMPLitude]?	Query current amplitude.		✓
:FUNCtion	Select source function.		✓
[:MODE] <name>	Select active source function. <name> = VOLTage | CURRent | TEMPerature | RESistance	TEMPerature	✓
[:MODE]?	Query active source function.		✓
:RESistance	Resistance function commands.		✓
:LCONstants	Set resistance PID loop constants.		
[:GAIN] <n>	Set resistance loop gain. <n> = 0 to 999.999 | MIN | MAX | DEF	20.0	
[:GAIN]?	Query resistance loop gain.		
:DERivative <n>	Set resistance derivative constant. <n> = 0 to 100 | MIN | MAX | DEF	0.0	
:DERivative?	Query resistance derivative constant.		
:INTegral <n>	Set resistance integral constant. <n> = 0 to 100 | MIN | MAX | DEF	0.6	
:INTegral?	Query resistance integral constant.		

Table 10-5 (cont.)
:SOURce[1] subsystem commands

Command	Description	Default Parameter	SCPI
:SOURce[1]			
:RESistance			
[:LEVel]			✓
[:IMMediate]			✓
[:AMPLitude] <n >	Set resistance level. <n> = 0.5 to 9.999e2 (Ω) for RTD or 100Ω thermistor, 5 to 9.999e3 (Ω) for 1kΩ thermistor, 50 to 8.000e4 (Ω) for 10Ω thermistor, 500 to 2.000e5 (Ω) for 100Ω thermistor | MIN | MAX | DEF	100 (RTD or 100Ω ther), 1e3 (1kΩ ther), 1e4 (10kΩ ther), or 1e5 (100kΩ ther)	✓
[:AMPLitude]?	Query resistance level.		✓
:PROTection	Set resistance protection.		✓
[:HIGH]	Upper resistance limit.		
[:LEVel] <n>	Set upper resistance limit. <n> 0.5 to 9.999e2 (Ω) for RTD or 100Ω thermistor, 5 to 9.999e3 (Ω) for 1kΩ thermistor, 50 to 8.000e4 (Ω) for 10kΩ thermistor, 500 to 2.000e5 (Ω) for 100kΩ thermistor | MIN (LOW:LEV) | MAX | DEF	200 (RTD or 100Ω ther), 2e3 (1kΩ ther), 2e4 (10kΩ ther), or 2e5 (100kΩ ther)	
[:LEVel]?	Query upper protection resistance limit.		
:TRIPped?	Query if upper limit is exceeded. (1 = yes; 0 = no).		
:LOW	Lower resistance limit.		
[:LEVel] <n>	Set lower resistance limit. <n> 0.5 to 9.999e2 (Ω) for RTD or 100Ω thermistor, 5 to 9.999e3 (Ω) for 1kΩ thermistor, 50 to 8.000e4 (Ω) for 10kΩ thermistor, 500 to 2.000e5 (Ω) for 100kΩ thermistor | MIN | MAX (HIGH:LEV) | DEF	50 (RTD or 100Ω ther), 5e2 (1kΩ ther), 5e3 (10kΩ ther), or 5e4 (100kΩ ther)	
[:LEVel]?	Query lower resistance limit.		
:TRIPped?	Query if lower limit exceeded. (1 = yes; 0 = no).		
:STATe <b>	Enable/disable resistance protection. (ON | OFF).	ON	✓
:STATe?	Query resistance protection state.		✓

Table 10-5 (cont.)
:SOURce[1] subsystem commands

Command	Description	Default Parameter	SCPI
:SOURce[1]			
:TEMPerature	Temperature commands.		✓
:LCONstants	Program temperature PID loop constants.		✓
[:GAIN] <n>	Set temperature gain. <n> = 0 to 999.999 | MIN | MAX | DEF	20.0	✓
[:GAIN]?	Query temperature gain.		✓
:DERivative <n>	Set temperature derivative constant. <n> = 0 to 100 | MIN | MAX | DEF	0.0	✓
:DERivative?	Query temperature derivative constant.		✓
:INTegral <n>	Set temperature integral constant. <n> = 0 to 100 | MIN | MAX | DEF	0.6	✓
:INTegral?	Query temperature integral constant.		✓
:PROTection	Set temperature protection.		✓
[:HIGH]	Upper temperature limit.		✓
[:LEVel] <n>	Set upper temperature limit. <n> = -50 to +250°C, 223.15 to 523.15K, -58 to 482°F | MIN | MAX | DEF	50.0°C, 323.15K, 122°F	✓
[:LEVel]?	Query upper protection temperature limit.		✓
:TRIPped?	Query if upper limit is exceeded. (1 = yes; 0 = no).		✓
:LOW	Lower temperature limit.	0.0	✓
[:LEVel] <n>	Set lower temperature limit. <n> = -50 to +250°C, 223.15 to 523.15K, -58 to 482°F | MIN | MAX | DEF	0°C, 273.15K, 32°F	✓
[:LEVel]?	Query lower protection temperature limit.		✓
:TRIPped?	Query if lower limit exceeded. (1 = yes; 0 = no).		✓
:STATe <b>	Enable/disable temperature protection. (ON | OFF)	ON	
:STATe?	Query temperature protection state.		
[:SPOint] <n>	Program temperature setpoint. <n> = -50 to +225°C, 223.15 to 498.15K, -58 to 437°F | MIN | MAX | DEF	25°C, 298.15K, 77°F	✓
[:SPOint]?	Query temperature setpoint.		✓

Table 10-5 (cont.)
:SOURce[1] subsystem commands

Command	Description	Default Parameter	SCPI
:SOURce[1]			✓
:STOLerance	Program setpoint tolerance parameters.		
[:PERCent] <n>	Percentage of range. <n> = 0.00-100.00 (%) | MIN | MAX | DEF	0.5	
[:PERCent]?	Query percent of range value.		
:POINts?	Query number of points currently within tolerance.		
:COUNt <n>	Number of points in tolerance window. <n> = 1-100 | MIN | MAX | DEF	5	
:COUNt?	Query number of points.		
:VOLTage	Voltage source commands.		✓
:LCONstants	Set voltage PID loop constants.		
[:GAIN] <n>	Program voltage loop gain. <n> = 0 to 999.999 | MIN | MAX | DEF	10.0	
[:GAIN]?	Query voltage loop gain.		
:DERivative <n>	Set voltage derivative constant. <n> = 0 to 100 | MIN | MAX | DEF	0.0	
:DERivative?	Query voltage derivative constant.		
:INTegral <n>	Set voltage integral constant. <n> = 0 to 100 | MIN | MAX | DEF	20.0	
:INTegral?	Query voltage integral constant.		
[:LEVel]			✓
[:IMMediate]			✓
[:AMPLitude] <n>	Set voltage amplitude. <n> = -10 to +10 (V) | MIN | MAX | DEF	0.0	✓
[:AMPLitude]?	Query voltage amplitude.		✓
:PROTection			✓
[:LEVel] <n>	Set voltage limit. <n> = 0.5 to +10.5 (V) | MIN | MAX | DEF	10.5	✓
[:LEVel]?	Query voltage limit.		✓
:TRIPped?	Query if limit exceeded. (1 = tripped; 0 = otherwise).		✓

Table 10-6
:SOURce2 subsystem commands

Command	Description	Default Parameter	SCPI
:SOURce2	Path to SOURce2 subsystem.		
:TTL			
[:LEVel] <NRf> or <NDN>	Specify the four bit Digital I/O port output value.	15	
[:LEVel]?	Query digital output line logic levels.		

Table 10-7
:STATus subsystem commands

Command	Description	Default Parameter	SCPI
:STATus		Note 1	✓
:MEASurement	Path to control measurement event registers.		
[:EVENt]?	Read the event register.[6]	Note 2	✓
:ENABle <NRf> | <NDN>	Program the enable register.	Note 3	✓
:ENABle?	Read the enable register.[6]		✓
:CONDition?	Read the condition register.[6]		✓
:OPERation	Path to control operation event registers.		✓
[:EVENt]?	Read the event register.[6]	Note 2	✓
:ENABle <NRf> | <NDN>	Program the enable register.	Note 3	✓
:ENABle?	Read the enable register.[6]		✓
:CONDition?	Read the condition register.[6]		✓
:QUEStionable	Path to control questionable event registers.		✓
[:EVENt]?	Read the event register.[6]	Note 2	✓
:ENABle <NRf> | <NDN>	Program the enable register.	Note 3	✓
:ENABle?	Read the enable register.[6]		✓
:CONDition?	Read the condition register.[6]		✓
:PRESet	Return status registers to default states.		✓
:QUEue	Path to access error queue.		
[:NEXT]?	Read the most recent error message.	Note 4	✓
:ENABle <list>	Specify error and status messages for queue.	Note 5	✓
:ENABle?	Read the enabled messages.		✓
:DISable <list>	Specify messages not to be placed in queue.	Note 5	
:DISable?	Read the disabled messages.		
:CLEar	Clears all messages from Error Queue.		

Notes:

1. Commands in this subsystem are not affected by *RST and :SYSTem:PREset. The effects of cycling power, *CLS and :STATus:PRESet, are explained by the following notes.
2. Event Registers — Power-up and *CLS clears all bits.:STATus:PRESet has no effect.
3. Enable Registers — Power-up and :STATus:PRESet clears all bits. *CLS has no effect. Accepts the SCPI 1995.0 mandated (non-decimal numeric) format (#H, #Q, or #B).
4. Error Queue — Power-up and *CLS clears all bits of the registers.
5. Error Queue Messages — Power-up clears list of messages. *CLS and :STATus:PRESet have no effect.
6. Register Query Commands — The format for the response messages (ASCII, hexadecimal, octal, or binary) depends on which data format is presently selected. See the :FORMat:SREGister command.

Table 10-8
:SYSTem subsystem commands

Command	Description	Default Parameter	SCPI
:SYSTem	Path to SYSTem subsystem.		
:KEY <n>	Simulate key press.		✓
:KEY?	Query the last pressed key.		✓
:PRESet	Return to :SYST:PRES (BENCH) defaults.		✓
:POSetup	Select power-on setup: RST, PRESet, or SAV0, 1, or 2.		
:POSetup?	Returns the power-on user setup.		
:VERSion?	Query rev level of SCPI standard (1996.0).		✓
:ERRor			✓
[:NEXT]?	Returns the latest error code, "Message."		
:ALL?	Return all errors.		
:COUNt?	Return how many errors.		
:CODE			
[:NEXt]?	Return the latest error code.		
:ALL?	Return all error codes.		
:CLEar	Clears messages from the Error Queue.		
:LFRequency	Sets the Line Frequency: 50 or 60Hz.	Note	
:LFRequency?	Returns the Line Frequency: 50 or 60Hz.		
:TIME			
RESet	Resets the Model 2510's absolute timestamp.		
:RSENse <b>	Enable (ON) or disable (OFF) 4-wire measurement. (RTD and thermistor sensors only.)	OFF	
:RSENse?	Query 4-wire measurement state.		
:GCONnect <b>	Connects (ON) or disconnects (OFF) input/output negative terminals to or from analog common.	OFF	
:GCONnect?	Query ground connect state.		
:LOCal	Take 2510 out of remote.		
:REMote	Put 2510 in remote.		
:RWLock	Enable local lockout.		

Note: The line frequency setting is not affected by *RST or :SYSTem:PRESet.

Table 10-9
Trigger subsystem commands

Command*	Description	Default Parameter	SCPI
:INITiate			
[:IMMediate]	Take instrument out of IDLE, begin taking readings.		✓
:ABORt	Turn output off, put instrument in IDLE.		✓

*Trigger commands are functional only in calibration mode. See Service Manual.

Table 10-10
:UNIT subsystem commands

Command	Description	Default Parameter	SCPI
:UNIT	Path to control temperature measurement units.		
:TEMPerature <name>	Program temperature units. <name> = C | CEL | F | FAR | K	Celsius	
:TEMPerature?	Query temperature units.		

DISPlay subsystem

The DISPlay subsystem is used to control the Model 2510 front panel two-line display via remote. The top, larger display has 20 characters, while the bottom, smaller display has 32 characters. Commands are summarized in Table 10-1.

Control display

:ATTRibutes?

:DISPlay[:WINDow[1]]:ATTRibutes? Query attributes; top display
:DISPlay:WINDow2:ATTRibutes? Query attributes; bottom display

Description This query command is used to determine which characters on the display are blinking and which are not. The response message provides that status of each character position for the specified display. The primary display consists of 20 characters and the secondary display consists of 32 characters.

1 = Character is blinking
0 = Character is not blinking

For example, assume the following menu is displayed with the SAVESETUP option blinking:

MAIN MENU
SAVESETUP COMMUNICATION CAL >

The response message for :disp:attr? (top display) will result in 20 zeroes as follows:

00000000000000000000

The response message for :disp:wind2:attr? (bottom display) will display ones at the character positions for SAVESETUP as follows:

11111111100000000000000000000000

Read display

:DATA?

:DISPlay[:WINDow[1]]:DATA? Read top display
:DISPlay:WINDow2:DATA? Read bottom display

Description These query commands are used to read what is currently being displayed on the top and bottom displays. After sending one of these commands and addressing the Model 2510 to talk, the displayed data (message or reading) will be sent to the computer.

Define :TEXT messages

:DATA <a>

:DISPlay[:WINDow[1]]:TEXT:DATA <a> — Define message; top display
:DISPlay:WINDow2:TEXT:DATA <a> — Define message; bottom display

Parameters

<a> = ASCII characters for message

Types:	String	'aa...a' or "aa...a"
	Indefinite Block	#0aa...a
	Definite Block	#XYaa...a
where	Y = number of characters in message:	
	Up to 20 for top display	
	Up to 32 for bottom display	
	X = number of digits that make up Y (1 or 2)	

Query :DATA? — Query the defined text message

Description

These commands define text messages for the display. A message can be as long as 20 characters for the top display, and up to 32 characters for the bottom display. A space is counted as a character. Excess message characters result in an error. (See "ASCII display values" below.)

An indefinite block message must be the only command in the program message or the last command in the program message. If you include a command after an indefinite block message (on the same line), it will be treated as part of the message and is displayed instead of executed.

:STATe <b>

:DISPlay[:WINDow[1]]:TEXT:STATe <b> — Control message; top display
:DISPlay:WINDow2:TEXT:STATe <b> — Control message; bottom display

Parameters

<b> =	0 or OFF	Disable text message for specified display
	1 or ON	Enable text message for specified display

Query :STATe? — Query state of message mode for specified display

Description

These commands enable and disable the text message modes. When enabled, a defined message is displayed. When disabled, the message is removed from the display.

GPIB Operation — A user defined text message remains displayed only as long as the instrument is in remote. Taking the instrument out of remote (by pressing the DISPLAY TOGGLE/LOCAL key or sending LOCAL 15) cancels the message and disables the text message mode.

RS-232 Operation — A user defined test message can be cancelled by sending the :SYSTem:LOCal command or pressing the LOCAL key.

ASCII display values

VAL	DISPLAY	VAL	DISPLAY	VAL	DISPLAY
000	invalid	048	0	096	` (open quote)
001	invalid	049	1	097	a
002	invalid	050	2	098	b
003	invalid	051	3	099	c
004	invalid	052	4	100	d
005	invalid	053	5	101	e
006	invalid	054	6	102	f
007	invalid	055	7	103	g
008	invalid	056	8	104	h
009	invalid	057	9	105	i
010	invalid	058	:	106	j
011	invalid	059	;	107	k
012	invalid	060	<	108	l
013	invalid	061	=	109	m
014	invalid	062	>	110	n
015	invalid	063	?	111	o
016	µ	064	@	112	p
017	±	065	A	113	q
018	W	066	B	114	r
019	×	067	C	115	s
020	leftflagbar	068	D	116	t
021	rightflagbar	069	E	117	u
022	fullflagbar	070	F	118	v
023	leftbar	071	G	119	w
024	rightbar	072	H	120	x
025	fullflagbar	073	I	121	y
026	¦	074	J	122	z
027	Ø	075	K	123	{
028	¨	076	L	124	|
029	Æ	077	M	125	}
030	selftest1	078	N	126	~
031	selftest2	079	O	127	box
032	space	080	P	128	space
033	!	081	Q	129	dot1
034	"	082	R	130	dot2
035	#	083	S	131	dot12
036	$	084	T	132	dot3
037	%	085	U	133	dot13
038	&	086	V	134	dot23
039	' (apostrophe)	087	W	135	dot123
040	(	088	X	136	dot4
041	)	089	Y	137	dot14
042	*	090	Z	138	dot24
043	+	091	[	139	dot124
044	, (comma)	092	\	140	dot34
045	-	093	]	141	dot134
046	.	094	^	142	dot234
047	/	095	_	143	dot1234

VAL	DISPLAY	VAL	DISPLAY	VAL	DISPLAY
144	selftest3	192	≥	240	ÿ
145	selftest4	193	≠	241	lightning
146	selftest5	194	≡	242	KI
147	selftest6	195	≈	243	box3
148	selftest7	196	∞	244	box4
149	selftest8	197	»	245	box5
150	selftest9	198	«	246	box6
151	selftest10	199	¿	247	box7
152	selftest11	200	¡	248	box8
153	selftest12	201	¢	249	box9
154	selftest13	202	£	250	tombstone
155	selftest14	203	¥	251	†
156	box1	204	Pt	252	↑
157	box2	205	*f*	253	↓
158	.5	206	Ç	254	→
159	¼	207	ç	255	←
160	^{0}	208	æ		
161	^{1}	209	Æ		
162	^{2}	210	â		
163	^{3}	211	ä		
164	^{4}	212	à		
165	^{5}	213	á		
166	^{6}	214	à		
167	^{7}	215	ª		
168	^{8}	216	Ä		
169	^{9}	217	Å		
170	α	218	ê		
171	β	219	ë		
172	γ	220	é		
173	δ	221	è		
174	ε	222	É		
175	η	223	î		
176	θ	224	ï		
177	λ	225	í		
178	π	226	ì		
179	ρ	227	ô		
180	σ	228	ö		
181	τ	229	ó		
182	φ	230	ò		
183	ω	231	º		
184	Γ	232	Ö		
185	Δ	233	û		
186	Σ	234	ü		
187	Φ	235	ú		
188	∩	236	ù		
189	∪	237	Ü		
190	÷	238	ñ		
191	≤	239	Ñ		

FORMat subsystem

The commands for this subsystem are used to select the data format for transferring instrument readings over the bus. These commands are summarized in Table 10-2.

Data format

[:DATA] <type>[,length]

:FORMat[:DATA] <type>[,<length>] Select data format

Parameters <type>[,<length>] =

ASCii	ASCII format
REAL,32	IEEE754 single precision format
SREal	IEEE754 single precision format

NOTE *<length> is not used for the ASCii or SREal parameters. It is optional for the REAL parameter. If you do not use <length> with the REAL parameter, the <length> defaults to 32 (single precision format).*

Query [:DATA]? Query data format

Description This command is used to select the data format for transferring readings over the bus. Only the ASCII format is allowed over the RS-232 interface. This command only affects the output of READ?, FETCh?, and MEASure[:<function>]?, over the GPIB. All other queries are returned in the ASCII format.

NOTE *Regardless of which data format for output strings is selected, the Model 2510 will only respond to input commands using the ASCII format.*

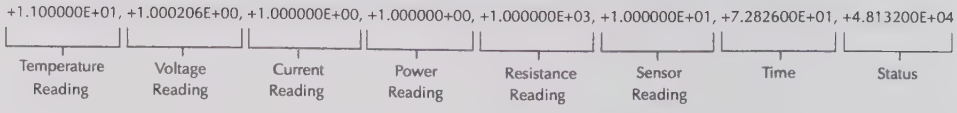
ASCII format

The ASCII data format is in a direct readable form for the operator. Most high-level computer languages easily convert ASCII mantissa and exponent to other formats. However, some speed is compromised to accommodate the conversion. Figure 10-1 shows an example ASCII string that includes all the data elements. See :ELEMents.

Figure 10-1 also shows the byte order of the data string. Data elements not specified by the :ELEMents command are simply not included in the string.

Keep in mind that the byte order can only be reversed for the binary format. See :BORDer.

Figure 10-1 ASCII data format



IEEE-754 single precision format

REAL,32, or SREal will select the binary IEEE-754 single precision data format. Figure 10-2 shows the normal byte order format for each data element. For example, if three valid elements are specified, the data string for each reading conversion is made up of three 4-byte data blocks. Note that the data string for each reading conversion is preceded by a 2-byte header that is the binary equivalent of an ASCII # sign and 0.

NOTE *This model terminates the binary data string with LF + EOI. If your program is set to terminate on CR and/or LF, data transfer may terminate prematurely because one or more data bytes has a CR or LF value. To avoid the problem, set your program to terminate on EOI only.*

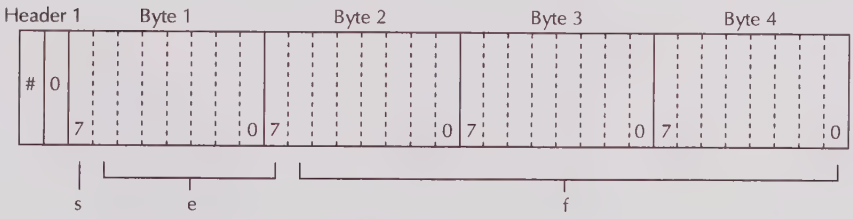


Figure 10-2 *IEEE-754 single precision data format (32 data bits)*

s = sign bit (0 = positive, 1 = negative)
e = exponent bits (8)
f = fraction bits (23)

Normal byte order shown. For swapped byte order, bytes sent in reverse order: Header, Byte 4, Byte 3, Byte 2, Byte 1.

The header and terminator are sent only once for each READ?

During binary transfers, never un-talk the Model 2510 until after the data is read (input) to the computer. Also, to avoid erratic operation, the readings of the data string (and terminator) should be acquired in one piece. The header (#0) can be read separately before the rest of the string.

The number of bytes to be transferred can be calculated as follows:

Bytes=2+(Rdgs × 4) + 1

where 2 is the number of bytes for the header (#0).

Rdgs is the product of the number of selected data elements, arm count, and trigger count.

4 is the number of bytes for each reading.

1 is the byte for the terminator.

Data elements

:ELEMents <item list>

:FORMat:ELEMents <item list> Specify data elements for data string

Parameters <item list> =

VOLTage	Includes voltage reading
CURRent	Includes current reading
POWer	Includes power reading
RESistance	Includes resistance reading
TEMPerature	Includes temperature reading
TSENsor	Includes sensor reading
TIME	Includes timestamp
STATus	Includes status information

NOTE *Each item in the list must be separated by a comma (i.e., :ELEMents, VOLTage, CURRent, RESistance).*

Query :ELEMents? Query elements in data string

Description This command is used to specify the elements to be included in the data string in response to the following queries:

:FETCh?
:READ?
:MEASure[:<function>]?

You can specify from one to all five elements. Each element in the list must be separated by a comma (,). These elements (shown in Figure 10-1) are explained as follows:

NOTE *An overflow reading reads as +9.9E37.*

VOLTage — This element provides the TEC voltage measurement.

CURRent — This element provides the TEC current measurement.

RESistance — This element provides the TEC resistance measurement.

TEMPerature — This element provides the TEC temperature measurement.

TSENSor — This element provides the sensor temperature measurement.

TIME — A timestamp is available to reference each group of readings to a point in time. The relative timestamp operates as a timer that starts at zero seconds when the instrument is turned on or when the relative timestamp is reset (:SYSTem:TIME:RESet). The timestamp for each reading sent over the bus is referenced, in seconds, to the start time.

STATus — Each and every measurement performed by the Model 2510 has a STATus word associated with it. This STATus word is accessible via the :FORMat:ELEMents STATus element. Internally, the Model 2510 maintains this STATus element as a 16-bit integer. Each bit within this integer

provides status information about a particular process associated with the construction of the measurement. However, when queried, the decimal equivalent is returned. The format of the STATus element is as follows:

Bit	Name	Description
0	VOLTAGE_OFLO	This bit is set to 1 when the voltage measurement was made in an over-range condition. Cleared to 0 otherwise.
1	CURRENT_OFLO	This bit is set to 1 when the current measurement was made in an over-range condition. Cleared to 0 otherwise.
2	PID_BIT_A	Encoded PID function enabled bit.
3	PID_BIT_B	Encoded PID function enabled bit. Measure PID bit encoding: Bit B / Bit A / Description 0 / 0 / Voltage Active 0 / 1 / Current Active 1 / 0 / Temperature Active 1 / 1 / Resistance Active
4	REMOTE_SENSE	1 if in 4-Wire Mode, 0 if 2-Wire
5	GND_CONNECT	1 = Connected; 0 = Disconnected
6	OPEN_LEAD	Set if open sensor lead condition exists.
7	SHORTED_LEAD	Set if shorted sensor lead condition exists.

Open-lead and shorted-lead conditions will be flagged under the following conditions:

Sensor	Open Lead	Shorted Lead
Thermistor	Temp. < -50°C	Temp. > 250°C
RTD	Temp. > 250°C	Temp. < -50°C
V-SS	Temp. < -50°C	-0.001V to +0.001V
I-SS	Temp. < -50°C	Temp. > 250°C

:SOURce2 <name>

:FORMat:SOURce2 <name> — Set SOUR2 and TTL response formats

Parameters

<name> =	ASCii	ASCII format
	HEXadecimal	Hexadecimal format
	OCTal	Octal format
	BINary	Binary format

Query :SOURce2? — Query response format

Description This command controls the response format for all SOUR2:TTL queries in a manner similar to formats set by the FORM:SREG command. See "SOURce2 subsystem" topics for details.

Byte order

:BORDer <name>

:FORMat:BORDer <name> — Specify binary byte order

Parameters

<name> =	NORMal	Normal byte order for binary formats
	SWAPped	Reverse byte order for binary formats

Query :BORDer? — Query byte order

Description This command is used to control the byte order for the IEEE-754 binary formats. For normal byte order, the data format for each element is sent as follows:

Byte 1 Byte 2 Byte 3 Byte 4 (Single precision)

For reverse byte order, the data format for each element is sent as follows:

Byte 4 Byte 3 Byte 2 Byte 1 (Single precision)

The "#0" Header is not affected by this command. The Header is always sent at the beginning of the data string for each measurement conversion.

The ASCII data format can only be sent in the normal byte order. The SWAPped selection is simply ignored when the ASCII format is selected.

NOTE *The SWAPped byte order must be used when transmitting binary data to any IBM PC compatible computer.*

Status register format

:SREGister <name>

:FORMat:SREGister <name> — Set data format for reading status registers

Parameters

<name> =		
	ASCii	Decimal format
	HEXadecimal	Hexadecimal format
	OCTal	Octal format
	BINary	Binary format

Query :SREGister? — Query format for reading status registers

Description Query commands are used to read the contents of the status event registers. This command is used to set the response message format for those query commands.

When a status register is queried, the response message is a value that indicates which bits in the register are set. For example, if bits B5, B4, B2, B1, and B0 of a register are set (110111), the following values will be returned for the selected data format:

ASCii	55	(decimal value)
HEXadecimal	#H37	(hexadecimal value)
OCTal	#Q67	(octal value)
BINary	#B110111	(binary value)

See Section 7, "Status structure and STATus subsystem" in this section for more information.

OUTPut subsystem

This subsystem is used to turn the source output on or off, and to control the Digital I/O port output enable line. These commands are summarized in Table 10-3.

Turn source on or off

[:STATe] <b>

:OUTPut[1][:STATe] <b> — Turn source on or off

Parameters

<b> =		
	0 or OFF	Turn source off (standby)
	1 or ON	Turn source on (operate)

Query :OUTPut? — Query state of source

Description This command is used to turn the source output on or off. Control or measurements cannot be performed while the source is off. Turning the source off places the Model 2510 in the idle state.

Output enable line control

:STATe <b>

:OUTPut[1]:ENABle[:STATe] <b> Control enable line

Parameters <b> = 0 or OFF Turn off enable line
1 or ON Turn on enable line

Query :STATe? Query state of enable

Description This command is used to activate or inactivate the hardware output enable line. When activated, the source cannot be turned on unless the output enable line (pin 8 of the rear panel Enable - Digital I/O connector) is pulled to a logic low state. When the enable line goes to a logic high state, the source turns off. See Section 5, "Digital I/O port and enable" for details using the enable line with a test fixture.

When disabled, the logic level on the enable line has no effect on the output state of the source.

:TRIPped?

:OUTPut[1]:ENABle:TRIPped?

Description This query command is used to determine if the output enable line state has been tripped. The tripped condition ("1") means that the source can be turned on (enable line at logic low level). A "0" will be returned if the source cannot be turned on (enable line at logic high level).

SENSe1 subsystem

The SENSe1 subsystem is used to configure and control the measurement functions of the Model 2510. Measurement functions include current, temperature, and resistance.

The commands for this subsystem are summarized in Table 10-4.

Current function

:PROTection[:LEVel] <n>

[:SENSe[1]]:CURRent[:DC]:PROTection[:LEVel] <n> Program current protection limit

Parameters <n> = 1.0 to +5.25A Set current protection limit
MINimum 1.0
MAXimum +5.25
DEFault 2.0

Query :PROTection[:LEVel]? Query current protection limit

Description This command programs the current protection limit. You can determine whether or not the limit has been exceeded by using the :TRIPped? query (below).

:TRIPped? <n>

[:SENSe[1]]:CURRent[:DC]:PROTection:TRIPped? Query if current limit exceeded

Description This command queries whether or not the current protection limit has been exceeded. A returned value of 1 = current limit exceeded; 0 = current limit not exceeded.

NOTE *The SOURce1 subsystem commands determine which function is controlled by the PID loop. All SENSe1 parameters are monitored continuously.*

Resistance function

:CURRent <n>

[:SENSe[1]]:RESistance:CURRent <n> Set RTD or thermistor sensor current

Parameters	<n> =	3e-6 to 2.5e-3	Set sensor current (3.3e-6, 1e-5, 3.33e-5, 1e-4, 8.333e-4, 2.5e-3)
		MINimum	3e-6
		MAXimum	2.5e-3
		DEFault	2.5e-3 (100Ω range, RTD) 8.333e-4 (1kΩ range) 1e-4 (10kΩ range) 3.33e-5 (100kΩ range)

Query :CURRent? Query sensor current

Description This command programs the temperature sensor (transducer) excitation current. It affects only thermistor and RTD sensor types. Nominal sensor currents are: 3.3e-6, 1e-5, 3.33e-5, 1e-4, 8.333e-4, and 2.5e-3.

:AUTO <b>

[:SENSe[1]]:RESistance:CURRent:AUTO <b> Enable/disable auto sensor current

Parameters	<b> =	1 or ON	Enable auto sensor current
		0 or OFF	Disable auto sensor current

Query :AUTO? Query auto sensor current state

Description This command enables or disables the auto sensor current mode. When the auto current mode is enabled, the sensor current is set to its default value.

:TRANsducer <name>

[:SENSe[1]]:RESistance:TRANsducer <name> Select resistance temperature sensor

Parameters

<name> =	THERmistor	Select thermistor sensor
	RTD	Select RTD sensor

Query :TRANsducer? Query sensor type

Description This command selects the temperature sensor for the resistance function. Only thermistor and RTD sensor types are available for the resistance function.

:RANGe <n>

[:SENSe[1]]:RESistance:THERmistor:RANGe <n> Set thermistor resistance range

Parameters

<n> =	0 to 3e5	Set thermistor resistance range (100, 1e3, 1e4, 1e5)
	MINimum	0
	MAXimum	3e5
	DEFault	1e4

Query :RANGe? Query sensor resistance range

Description This command programs the temperature sensor resistance range. It affects only thermistor sensor types. Nominal ranges are: 100, 1e3, 1e4, and 1e5.

Temperature function

:CURRent <n>

[:SENSe[1]]:TEMPerature:CURRent <n> Set RTD or thermistor sensor current

Parameters

<NRf> =	3e-6 to 2.5e-3	Set sensor current (3.3e-6, 1e-5, 3.33e-5, 1e-4, 8.333e-4, 2.5e-3)
	MINimum	3e-6
	MAXimum	2.5e-3
	DEFault	2.5e-3 (100Ω range, RTD) 8.333e-4 (1kΩ range) 1e-4 (10kΩ range) 3.33e-5 (100kΩ range)

Query :CURRent? Query sensor current

Description This command programs the temperature sensor (transducer) excitation current. It affects only thermistor and RTD sensor types. Nominal sensor currents are: 3.3e-6, 1e-5, 3.33e-5, 1e-4, 8.333e-4, and 2.5e-3.

:AUTO <b>

[:SENSe[1]]:TEMPerature:CURRent:AUTO <b> — Enable/disable auto sensor current

Parameters

<b> =	1 or ON	Enable auto sensor current
	0 or OFF	Disable auto sensor current

Query :AUTO? — Query auto sensor current state

Description This command enables or disables the auto sensor current mode. When the auto current mode is enabled, the sensor current is set to its default value.

:TRANsducer <name>

[:SENSe[1]]:TEMPerature:TRANsducer <name> — Select temperature function sensor

Parameters

<name> =	THERmistor	Select thermistor sensor
	RTD	Select RTD sensor
	VSS	Voltage type solid-state sensor
	ISS	Current type solid-state sensor

Query :TRANsducer? — Query sensor type

Description This command select the temperature sensor for the temperature function. All four sensor types are available for use with the temperature function.

RTD sensor parameters

:TYPE <name>

[:SENSe[1]]:TEMPerature:RTD:TYPE <name> — Select RTD sensor type

Parameters

<name> =	PT385	Select PT385 RTD sensor
	PT3916	Select PT3916 RTD sensor
	PT100	Select PT100 RTD sensor
	D100	Select D100 RTD sensor
	F100	Select F100 RTD sensor
	USER	Select USER sensor

Query :TYPE? — Query RTD sensor type

Description This command selects the RTD temperature sensor type for the temperature and function. When USER is selected, α, β, and δ must also be user programmed (see below).

:ALPHa <n>

[:SENSe[1]]:TEMPerature:RTD:ALPHa <n> Program RTD sensor alpha

Parameters	<n> =	0 to 0.01	RTD α coefficient
		MINimum	0
		MAXimum	0.01
		DEFault	0.00385

Query :ALPHa? Query RTD α coefficient

Description This command programs the RTD temperature sensor α coefficient for the USER type RTD sensor.

NOTE *See Section 4, "Sensor coefficients."*

:BETA <n>

[:SENSe[1]]:TEMPerature:RTD:BETA <n> Program RTD sensor beta

Parameters	<n> =	0 to 1	RTD β coefficient
		MINimum	0
		MAXimum	1
		DEFault	0.111

Query :BETA? Query RTD β coefficient

Description This command programs the RTD temperature sensor β coefficient for the USER type RTD sensor.

:DELTa <n>

[:SENSe[1]]:TEMPerature:RTD:DELTa <n> Program RTD sensor delta

Parameters	<n> =	0 to 5	RTD δ coefficient
		MINimum	0
		MAXimum	5
		DEFault	1.507

Query :DELTa? Query RTD δ coefficient

Description This command programs the RTD temperature sensor δ coefficient for the USER type RTD sensor.

Thermistor sensor parameters

:RANGe <n>

[:SENSe[1]]:TEMPerature:THERmistor:RANGe <n> Set thermistor resistance range

Parameters	<n> =	0 to 3e5	Set thermistor resistance range (100, 1e3, 1e4, 1e5)
		MINimum	0
		MAXimum	3e5
		DEFault	1e4

Query :RANGe? Query sensor resistance range

Description This command programs the temperature sensor resistance range. It affects only thermistor sensor types. Nominal ranges are: 100, 1e3, 1e4, and 1e5.

:A <n>

[:SENSe[1]]:TEMPerature:THERmistor:A <n> Program thermistor sensor A value

Parameters	<n> =	-10 to +10	thermistor A coefficient
		MINimum	-10
		MAXimum	+10
		DEFault	1.12396e-3

Query :A? Query thermistor A coefficient

Description This command programs the thermistor temperature sensor A coefficient.

NOTE *See Section 4, "Sensor coefficients."*

:B <n>

[:SENSe[1]]:TEMPerature:THERmistor:B <n> Program thermistor sensor B value

Parameters	<n> =	-10 to +10	thermistor B coefficient
		MINimum	-10
		MAXimum	+10
		DEFault	2.30093e-4

Query :B? Query thermistor B coefficient

Description This command programs the thermistor temperature sensor B coefficient.

:C <n>

[:SENSe[1]]:TEMPerature:THERmistor:C <n> Program thermistor sensor C value

Parameters	<n> =	-10 to +10	thermistor C coefficient
		MINimum	-10
		MAXimum	+10
		DEFault	7.06351e-8

Query :C? Query thermistor C coefficient

Description This command programs the thermistor temperature sensor C coefficient.

Solid-state sensor parameters

[:GAIN] <n>

[:SENSe[1]]:TEMPerature:VSS[:GAIN] <n> Program voltage sensor gain
[:SENSe[1]]:TEMPerature:ISS[:GAIN] <n> Program current sensor gain

Parameters	:VSS <n> =	0 to 9.999e-2	VSS solid-state sensor gain (V/K)
		MINimum	0
		MAXimum	9.999e-2
		DEFault	1.0e-2
	:ISS <n> =	0 to 9.999e-4	ISS solid-state sensor gain (A/K)
		MINimum	0
		MAXimum	9.999e-4
		DEFault	1.0e-6

Query [:GAIN]? Query solid-state sensor gain

Description These commands program the solid-state sensor gain values. :VSS sets voltage type sensor gain, and :ISS sets type current sensor gain.

:OFFSet <n>

[:SENSe[1]]:TEMPerature:VSS:OFFSet <n> Program voltage sensor offset
[:SENSe[1]]:TEMPerature:ISS:OFFSet <n> Program current sensor offset

Parameters	:VSS <n> =	0 to 999.999	VSS solid-state sensor offset (K)
		MINimum	0
		MAXimum	999.999
		DEFault	0
	:ISS <n> =	0 to 999.999	ISS solid-state sensor offset (K)
		MINimum	0
		MAXimum	999.999
		DEFault	0

Query :OFFSet? Query solid-state sensor offset

Description These commands program the solid-state sensor offset values. :VSS sets voltage type sensor offset, and :ISS sets current type sensor offset.

SOURce[1] subsystem

The SOURce[1] subsystem provides the most important Model 2510 control functions. This subsystem consists of a closed-loop PID (Proportional, Integral, and Derivative) to control temperature, current, voltage, and resistance. The closed-loop response of the system is dependent upon the settings of the proportional, integral, and derivative gain constants. The PID maintains balance in a closed-loop system by changing the control variable in response to set point changes (See Section 4, "PID control concepts"). This balance is accomplished by feeding the error, which is the difference between the set point and the process variable, to the PID controller, and the result of the PID calculation drives the process variable towards the set point. The Model 2510 can be configured as a 50W voltage source capable of delivering ±10V @ ±5A or a 50W current source capable of delivering ±5A @ ±10V.

Table 10-5 summarizes SOURce[1] subsystem commands.

Control source output-off

[:IMMediate]

:SOURce[1]:CLEar[:IMMediate] Turn source output off

Description This command is used to turn off the source output. The output will turn off after all programmed control-measure operations are completed. The instrument returns to the idle state when ABORT is sent. This command performs the same function as :OUTPut OFF.

Select source function

:FUNCtion[:MODE] <name>

:SOURce[1]:FUNCtion[:MODE] <name> Select source function

Parameters	<name> =	VOLTage	Select voltage source function
		CURRent	Select current source function
		TEMPerature	Select temperature function
		RESistance	Select resistance function

Query :FUNCtion[:MODE]? Query current amplitude

Description This command selects the Model 2510 source function in the same manner as the four front panel function keys (T, V, I, and R). System control depends upon correct programming of PID constants (see below).

Current function

[:AMPLitude] <n>

:SOURce[1]:CURRent[:LEVel][:IMMediate][:AMPLitude] <n> Program current amplitude

Parameters

<n> =	-5 to +5A	Set current amplitude
	MINimum	-5
	MAXimum	+5
	DEFault	0.0

Query [:AMPLitude]? Query current amplitude

Description This command programs the current amplitude (setpoint) for the current source function. The current amplitude can be set over a range of ±5A but cannot exceed the current protection limit.

:LCONstants

:SOURce[1]:CURRent:LCONstants[:GAIN] <n> Program proportional constant
:SOURce[1]:CURRent:LCONstants:DERvative <n> Program derivative constant
:SOURce[1]:CURRent:LCONstants:INTegral <n> Program integral constant

Parameters

:[GAIN] <n> =	0 to 999.999	Set proportional constant value
	MINimum	0
	MAXIMUM	999.999
	DEFault	10.0
:DERivative <n>=	0 to 100	Set derivative constant value
	MINimum	0
	MAXimum	100
	DEFault	0.0
:INTegral <n> =	0 to 100	Set integral constant value
	MINimum	0
	MAXimum	100
	DEFault	20.0

Query

[:GAIN]?	Query proportional constant
:DERivative?	Query derivative constant
:INTegral?	Query integral constant

Description These commands program the current function PID loop constants, which must be set to appropriate values for suitable control characteristics. See Section 4, "PID control concepts" for information on choosing appropriate values.

Resistance function

[:AMPLitude] <n>

:SOURce[1]:RESistance[:LEVel][:IMMediate][:AMPLitude] <n> Program resistance value

Parameters

<n> =		
	0.5 to 9.999e2Ω	RTD or 100Ω thermistor
	5 to 9.999e3Ω	1kΩ thermistor
	50 to 8.000e4Ω	10kΩ thermistor
	500 to 2.000e5Ω	100kΩ thermistor
	MINimum	0.5Ω (RTD or 100Ω thermistor)
		5Ω (1kΩ thermistor)
		50Ω (10kΩ thermistor)
		500Ω (100kΩ thermistor)
	MAXimum	9.999e2Ω (RTD or 100Ω thermistor)
		9.999e3Ω (1kΩ thermistor)
		8.000e4Ω (10kΩ thermistor)
		2.000e5Ω (100kΩ thermistor)
	DEFault	100Ω (RTD or 100Ω thermistor)
		1e3Ω (1kΩ thermistor)
		1e4Ω (10kΩ thermistor)
		1e5Ω (100kΩ thermistor)

Query [:AMPLitude]? Query resistance amplitude

Description This command programs the amplitude (setpoint) for the resistance function. The amplitude can be set over a range that depends on sensor type. The value cannot exceed the protection limits.

:LCONstants

:SOURce[1]:RESistance:LCONstants[:GAIN] <n> Program proportional constant
:SOURce[1]:RESistance:LCONstants:DERvative <n> Program derivative constant
:SOURce[1]:RESistance:LCONstants:INTegral <n> Program integral constant

Parameters

[:GAIN] <n> =		
	0 to 999.999	Set proportional constant value
	MINimum	0
	MAXimum	999.999
	DEFault	20.0
:DERivative <n> =	0 to 100	Set derivative constant value
	MINimum	0
	MAXimum	100
	DEFault	0.0
:INTegral <n> =	0 to 100	Set integral constant value
	MINimum	0
	MAXimum	100
	DEFault	0.6

Query

[:GAIN]?	Query proportional constant
:DERivative?	Query derivative constant
:INTegral?	Query integral constant

Description These commands program the resistance function PID loop constants, which must be set to appropriate values for suitable control characteristics. See Section 4, "PID control concepts" for information on choosing appropriate values.

:PROTection

:SOURce[1]:RESistance:PROTection[:HIGH][:LEVel] <n>	Program resistance high limit
:SOURce[1]:RESistance:PROTection[:HIGH]:TRIPped?	Query if high limit exceeded
:SOURce[1]:RESistance:PROTection:LOW[:LEVel] <n>	Program resistance low limit
:SOURce[1]:RESistance:PROTection:LOW:TRIPped?	Query if low limit exceeded
:SOURce[1]:RESistance:PROTection:STATe <b>	Enable/disable protection

Parameters

[:HIGH] <n> =	0.5 to 9.999e2Ω	RTD or 100Ω thermistor
	5 to 9.999e3Ω	1kΩ thermistor
	50 to 8.000e4Ω	10kΩ thermistor
	500 to 2.000e5Ω	100kΩ thermistor
	MINimum	0.5Ω (RTD or 100Ω thermistor)
		5Ω (1kΩ thermistor)
		50Ω (10kΩ thermistor)
		500Ω (100kΩ thermistor)
	MAXimum	9.999e2Ω (RTD or 100Ω thermistor)
		9.999e3Ω (1kΩ thermistor)
		8.000e4Ω (10kΩ thermistor)
		2.000e5Ω (100kΩ thermistor)
	DEFault	200Ω (RTD or 100Ω thermistor)
		2e3Ω (1kΩ thermistor)
		2e4Ω (10kΩ thermistor)
		2e5Ω (100kΩ thermistor)
:LOW <n> =	0.5 to 9.999e2Ω	RTD or 100Ω thermistor
	5 to 9.999e3Ω	1kΩ thermistor
	50 to 8.000e4Ω	10kΩ thermistor
	500 to 2.000e5Ω	100kΩ thermistor
	MINimum	0.5Ω (RTD or 100Ω thermistor)
		5Ω (1kΩ thermistor)
		50Ω (10kΩ thermistor)
		500Ω (100kΩ thermistor)
	MAXimum	9.999e2Ω (RTD or 100Ω thermistor)
		9.999e3Ω (1kΩ thermistor)
		8.000e4Ω (10kΩ thermistor)
		2.000e5Ω (100kΩ thermistor)
	DEFault	50Ω (RTD or 100Ω thermistor)
		5e2Ω (1kΩ thermistor)
		5e3Ω (10kΩ thermistor)
		5e4Ω (100kΩ thermistor)
:STATe <b> =	ON | OFF	Enable/disable protection
	ON | 1	Enable protection
	OFF | 0	Disable protection

Query

[:HIGH][:LEVel]?	Query high limit
[:HIGH]:TRIPped?	Query if high limit exceeded (1 = exceeded)
:LOW[:LEVel]?	Query low limit
:LOW:TRIPped?	Query if low limit exceeded (1 = exceeded)
:STATe?	Query if protection is enabled.

Description The :HIGH and :LOW commands program the high and low limits respectively for the resistance function, while the :STATe command enables and disables resistance function limits. The two :TRIPPed? queries return a value of 1 if the corresponding limit is exceeded or a value of 0 if it is not exceeded. :STATe? returns a value of 1 if protection is enabled. Note that the high limit cannot be lower than the low limit, and the low limit cannot be higher than the high limit.

Temperature function

[:SPOint] <n>

:SOURce[1]:TEMPerature:[SPOint] <n> Program temperature setpoint

Parameters

<n> =	-50 to +225°C	Set temperature setpoint
	223.15 to 498.15K	
	-58 to 437°F	
	MINimum	-50°C
		223.15K
		-58°F
	MAXimum	+225°C
		498.15K
		437°F
	DEFault	+25°C
		298.15K
		77°F

Query [:SPOint]? Query temperature setpoint

Description This command programs the temperature setpoint for the temperature function. This value is the temperature at which the Model 2510 will maintain system temperature when the temperature function is enabled. The setpoint cannot exceed the temperature protection limits.

:LCONstants

:SOURce[1]:TEMPerature:LCONstants[:GAIN] <n> Program proportional constant
:SOURce[1]:TEMPerature:LCONstants:DERvative <n> Program derivative constant
:SOURce[1]:TEMPerature:LCONstants:INTegral <n> Program integral constant

Parameters

[:GAIN] <n> =	0 to 999.999	Set proportional constant value
	MINimum	0
	MAXimum	999.999
	DEFault	20.0
:DERivative <n> =	0 to 100	Set derivative constant value
	MINimum	0
	MAXimum	100
	DEFault	0.0
:INTegral <n> =	0 to 100	Set integral constant value
	MINimum	0
	MAXimum	100
	DEFault	0.6

Query

[:GAIN]?	Query proportional constant
:DERivative?	Query derivative constant
:INTegral?	Query integral constant

Description

These commands program the temperature function PID loop constants, which must be set to appropriate values for suitable control characteristics. See Section 4, "PID control concepts" for information on choosing appropriate values.

:PROTection

:SOURce[1]:TEMPerature:PROTection[:HIGH][:LEVel] <n>	Program temperature high limit
:SOURce[1]:TEMPerature:PROTection[:HIGH]:TRIPped?	Query if high limit exceeded
:SOURce[1]:TEMPerature:PROTection:LOW[:LEVel] <n>	Program temperature low limit
:SOURce[1]:TEMPerature:PROTection:LOW:TRIPped?	Query if low limit exceeded
:SOURce[1]:TEMPerature:PROTection:STATe <b>	Enable/disable protection

Parameters

[:HIGH] <n> =	-50 to +250°C	Set high limit value
	223.15 to 523.15K	
	-58 to 482°F	
	MINimum	-50°C
		223.15K
		-58°F
	MAXimum	+250°C
		523.15K
		482°F
	DEFault	+50°C
		323.15K
		122°F
:LOW <n> =	-50 to +250°C	Set low limit value
	223.15 to 523.15K	
	-58 to 482°F	
	MINimum	-50°C
		223.15K
		-58°F
	MAXimum	+250
		523.15K
		482°F
	DEFault	0°C
		273.15K
		32°F
:STATe <b> =	ON | OFF	Enable/disable protection
	ON | 1	Enable protection
	OFF | 0	Disable protection

Query

[:HIGH][:LEVel]?	Query high limit
[:HIGH]:TRIPped?	Query if high limit exceeded (1 = exceeded)
:LOW[:LEVel]?	Query low limit
:LOW:TRIPped?	Query if low limit exceeded (1 = exceeded)
:STATe?	Query is protection enabled (1 = enabled)

Description

The :HIGH and :LOW commands program the high and low limits respectively for the temperature function, while the :STATe command enables and disables temperature function limits. The two ::TRIPPed? queries return a value of 1 if the corresponding limit is exceeded or a value of 0 if it is not exceeded. The high limit cannot be set lower than the low limit, and the low limit cannot be set higher than the high limit.

Voltage function

[:AMPLitude] <n>

:SOURce[1]:VOLTage[:LEVel][:IMMediate][:AMPLitude] <n> Program voltage source value

Parameters	<n> =	-10 to +10V	Set voltage source value
		MINimum	-10
		MAXimum	+10
		DEFault	0.0

Query [:AMPLitude]? Query voltage source value

Description This command programs the amplitude (setpoint) for the voltage source. The voltage amplitude can be set over a range of ±10V, but the value cannot exceed the voltage protection limit.

:LCONstants

:SOURce[1]:VOLTage:LCONstants[:GAIN] <n> Program proportional constant
:SOURce[1]:VOLTage:LCONstants:DERvative <n> Program derivative constant
:SOURce[1]:VOLTage:LCONstants:INTegral <n> Program integral constant

Parameters	[:GAIN] <n> =	0 to 999.999	Set proportional constant value
		MINimum	0
		MAXimum	999.999
		DEFault	10.0
	:DERivative <n> =	0 to 100	Set derivative constant value
		MINimum	0
		MAXimum	100
		DEFault	0.0
	:INTegral <n> =	0 to 100	Set integral constant value
		MINimum	0
		MAXimum	
		DEFault	20.0

Query
[:GAIN]? Query proportional constant
:DERivative? Query derivative constant
:INTegral? Query integral constant

Description These commands program the voltage function PID loop constants, which must be set to appropriate values for suitable control characteristics. See Section 4, "PID control concepts" for information on choosing appropriate values.

:PROTection

:SOURce[1]:VOLTage:PROTection[:LEVel] <n> Program voltage limit
:SOURce[1]:VOLTage:PROTection:TRIPped? Query if voltage limit exceeded

Parameters

<n> =	0.5 to +10.5V	Set voltage limit value
	MINimum	0.5
	MAXimum	+10.5
	DEFault	+10.5

Query

[:LEVel[?	Query voltage limit
:TRIPped?	Query if voltage limit exceeded (1 = exceeded)

Description The :LEVel command programs the voltage limit for the voltage function. The :TRIPPed? query returns a value of 1 if the voltage limit is exceeded or a value of 0 if it is not exceeded. These protection commands are active only for the voltage function and are absolute. As a result, there are no high/low limits or :STATe commands since voltage protection is always active.

Setpoint tolerance

[:PERCent] <n>

:SOURce[1]:STOLerance[:PERCent] <n> Program tolerance window

Parameters

<n> =	0 to 100%	Program tolerance (% of range)
	MINimum	0
	MAXimum	100
	DEFault	0.5

Query

[:PERCent]?	Query percent tolerance.

Description This command is used to program the percentage of range for the setpoint tolerance. Ranges depend on selected function as follows:

Function	Range
Temperature	275°C 275K 495°F
Voltage	20V
Current	10A
Resistance	1MΩ (100kΩ thermistor) 100kΩ (10kΩ thermistor) 10kΩ (1kΩ thermistor) 1kΩ (100Ω thermistor or RTD)

NOTE *Bit 12 of the Measurement Event Register will be set when the setpoint is within the percent tolerance for the programmed number of readings (see below). See Section 7 for status register details.*

:COUNt <n>

:SOURce[1]:STOLerance:COUNt <n> — Set number of readings for tolerance

Parameters	<n> =	1 to 100	Number of readings
		MINimum	1
		MAXimum	100
		DEFault	5

Query :COUNt? — Query number of readings.

Description This command is used to program how many readings should be within the tolerance window programmed with the :PERCent command for the set-point to be considered to be within tolerance.

:POINts?

:SOURce[1]:STOLerance:POINts? — Query number of readings within window

Description This command queries the number of readings that have currently been within the programmed tolerance window.

SOURce2 subsystem

The SOURce 2 subsystem command is used to set the logic level of the digital output lines, as summarized in Table 10-6.

[:LEVel] <NRf> | <NDN>

:SOURce2:TTL[:LEVel] <NRf> | <NDN> — Set digital output pattern

Parameters	<NRf> =	0 to 15	Decimal format
	<NDN>	#Bx	Binary format: x = 0000 to 1111
		#Hx	Hexadecimal format: x = 0 to F
		#Qx	Octal format: x = 0 to 17

Query :TTL? — Query digital output value

Description This command is used to set the logic levels of the output lines of the Digital I/O port. When set high, the specified output line will be at approximately +5V. When set low, the output line will be at 0V. The four output lines are internally pulled up to V_{CC} and are capable of sinking 500mA and sourcing 2mA. These lines are also capable of switching external voltages as high as 33V.

Use the following table to determine the parameter value for the desired decimal digital output pattern:

OUT 4	OUT 3	OUT 2	OUT 1	Decimal value*
L	L	L	L	0
L	L	L	H	1
L	L	H	L	2
L	L	H	H	3
L	H	L	L	4
L	H	L	H	5
L	H	H	L	6
L	H	H	H	7
H	L	L	L	8
H	L	L	H	9
H	L	H	L	10
H	L	H	H	11
H	H	L	L	12
H	H	L	H	13
H	H	H	L	14
H	H	H	H	15

L = Low (Gnd)
H = High (>+3V)

STATus subsystem

The STATus subsystem is used to control the status registers of the Model 2510. The commands in this subsystem are summarized in Table 10-7.

NOTE *These registers and the overall status structure are fully explained in Section 7.*

Read event registers

[:EVENt]?

:STATus:MEASurement[:EVENt]? Read Measurement Event Register
:STATus:QUEStionable[:EVENt]? Read Questionable Event Register
:STATus:OPERation[:EVENt]? Read Operation Event Register

Description These query commands are used to read the contents of the status event registers. After sending one of these commands and addressing the Model 2510 to talk, a value is sent to the computer. This value indicates which bits in the appropriate register are set.

Program event enable registers

:ENABle <NDN> or <NRf>

:STATus:MEASurement:ENABle <NDN> or <NRf> — Program Measurement Event Enable Register
:STATus:QUEStionable:ENABle <NDN> or <NRf> — Program Questionable Event Enable Register
:STATus:OPERation:ENABle <NDN> or <NRf> — Program Operation Event Enable Register

Parameters

<NDN> =	#Bxx...x	Binary format (each x = 1 or 0)
	#Hx	Hexadecimal format (x = 0 to FFFF)
	#Qx	Octal format (x = 0 to 177777)
<NRf> =	0 to 65535	Decimal format

Query :ENABle? — Read an enable register

Description These commands are used to program the enable registers of the status structure. The binary equivalent of the parameter value that is sent determines which bits in the register gets set. See Section 7 for details.

Read condition registers

:CONDition?

:STATus:MEASurement:CONDition? — Read Measurement Condition
:STATus:QUEStionable:CONDition? — Read Questionable Register
:STATus:OPERation:CONDition? — Read Operation Condition

Description These query commands are used to read the contents of the condition registers.

Select default conditions

:PRESet

:STATus:PRESet — Return registers to default conditions

Description When this command is sent, the following SCPI event registers are cleared to zero (0):

1. Operation Event Enable Register.
2. Event Enable Register.
3. Measurement Event Enable Register.

NOTE *The Standard Event Register is not affected by this command.*

Error queue

[:NEXT]?

:STATus:QUEue[:NEXT]? Read Error Queue

Description As error and status messages occur, they are placed into the Error Queue. This query command is used to read those messages. See Appendix B for a list of messages.

NOTE *The :STATus:QUEue[:NEXT]? query command performs the same function as the :SYSTem:ERRor? query command. See ":SYSTem subsystem."*

:CLEar

:STATus:QUEue:CLEar Clear Error Queue

Description This action command is used to clear the Error Queue of messages.

ENABle <list>

:STATus:QUEue:ENABle <list> Enable messages for Error Queue

Parameters <list> = (numlist)

where numlist is a specified list of messages that you wish to enable for the Error (must be in parentheses).

Query :ENABle? Query list of enabled messages

Description On power-up, all error messages are enabled and will go into the Error Queue as they occur. Status messages are not enabled and will not go into the queue. This command is used to specify which messages you want enabled. Messages not specified will be disabled and prevented from entering the queue.

DISable <list>

:STATus:QUEue:DISable <list> Disable messages for Error Queue

Parameters <list> = (numlist)

where numlist is a specified list of messages that you wish to disable for the Error Queue in parentheses.

Query :DISable? Query list of disabled messages

Description On power-up, all error messages are enabled and will go into the Error Queue as they occur. Status messages are not enabled and will not go into the queue. This command is used to specify which messages you want disabled. Disabled messages are prevented from going into the Error Queue.

SYSTem subsystem

The SYSTem subsystem contains miscellaneous commands that are summarized in Table 10-8.

Default conditions

:PRESet

:SYSTem:PRESet — Return to :SYSTem:PRESet defaults

Description This command returns the instrument to states optimized for front panel operation. :SYSTem:PRESet defaults are listed in Tables 10-1 through 10-10, and are the same as front panel BENCH defaults (See Section 1).

:POSetup

:SYSTem:POSetup <name> — Program power-on defaults

Parameters	<name> =	RST	Power-up to *RST defaults
		PRESet	Power-up to :SYSTem:PRESet defaults
		SAV0	Power-up to setup stored at memory location 0
		SAV1	Power-up to setup stored at memory location 1
		SAV2	Power-up to setup stored at memory location 2
Query	:POSetup?		Query power-on setup

Description This command is used to select the power-on defaults. With RST selected, the instrument powers up to the *RST default conditions. With PRES selected, the instrument powers up to the :SYStem:PRESet default conditions. Default conditions are listed in the Tables 10-1 through 10-10.

With the SAV0-2 parameters specified, the instrument powers-on to the setup that is saved in the specified location using the *SAV command.

Select power line frequency setting

:LFRequency <freq>

:SYSTem:LFRequency <freq> — Select line frequency

Parameters	<freq> =	50	50Hz setting
		60	60Hz setting
Query	:LFRequency?		Query line frequency selection.

Description Use this command to manually select the line frequency setting (50 or 60Hz). The line frequency setting should match the power line frequency, or readings may be noisy.

Error queue

NOTE *See Section 7 for details on the error queue.*

[:NEXT]?

:SYSTem:ERRor[:NEXT]? Read oldest error (code and message)

Description As error and status messages occur, they are placed in the Error Queue. The Error Queue is a first-in, first-out (FIFO) register that can hold up to 10 messages. After sending this command and addressing the Model 2510 to talk, the oldest message is sent to the computer and is then removed from the queue.

NOTE *The :STATus:Queue? command performs the same function as :SYSTem:ERRor[:NEXT]?. See "STATus subsystem."*

:ALL?

:SYSTem:ERRor:ALL? Read all errors (codes and messages)

Description This query command is similar to the [:NEXT]? command except that all messages in the Error Queue are sent to the computer when the Model 2510 is addressed to talk. All messages are removed from the queue.

:COUNt?

:SYSTem:ERRor:COUNt? Return the number of errors

Description After sending this command and addressing the Model 2510 to talk, a decimal number will be sent to the computer. That is the number of messages in the Error Queue.

:CODE[:NEXT]?

:SYSTem:ERRor:CODE[:NEXT]? Read oldest error (code only)

Description This command is identical to the [:NEXT]? command, except only the code is returned. The message itself is not returned. The error is cleared from the queue.

:CODE:ALL?

SYSTem:ERRor:CODE:ALL? Read all errors (codes only)

Description This query command is identical to the :SYSTem:ERRor:ALL? command, except only the codes are returned. The actual messages are not returned. All errors are cleared from the queue.

:CLEar

:SYSTem:CLEar — Clear Error Queue

Description This action command is used to clear the Error Queue of messages.

Simulate key presses

:KEY <NRf>

:SYSTem:KEY <NRf> — Simulate key-press

Parameters

<NRf> =	
4	Right arrow key
5	V
6	CONFIG
12	I
13	MENU
14	EXIT
16	TOGGLE
19	R
20	Up arrow key
21	ENTER
24	OUTPUT
27	Down arrow key
28	Left arrow key
29	T

Query :KEY? — Query last "pressed" key.

Description This command is used to simulate front panel key presses. For example, to select the MENU key you can send the following command to simulate pressing the menu key:

:syst:key 21

The parameter listing above provides the key-press code in numeric order.

The queue for the :KEY? query command can only hold one key-press. When :KEY? is sent over the bus, and the Model 2510 is addressed to talk, the key-press code number for the last key pressed (either physically or with :KEY) is sent to the computer.

The key-press code number for the last key pressed (either physically or with :key) is sent to the computer.

Read version of SCPI standard

:VERSion?

:SYSTem:VERSion? — Read SCPI version

Description This query command is used to read the version of the SCPI standard being used by the Model 2510. Example code:

1996.0

The above response message indicates the version of the SCPI standard.

Reset timestamp

:RESet

:SYSTem:TIME:RESet — Reset timestamp

Description This action command is used to reset the absolute timestamp to 0 seconds. The timestamp also resets to 0 seconds every time the Model 2510 is turned on. Note that time information can be returned with the :READ? and :MEAS? queries by specifying a time element with the :FORMat:ELEMents command (see "Format subsystem").

2-wire/4-wire sense mode

:RSENse <b>

:SYSTem:RSENse <b> — Control 4-wire/2-wire sensing

Parameters	<b> =	1 or ON	Enable 4-wire sensor sense mode
		0 or OFF	Disable 4-wire sensing (2-wire mode)
Query	:RSENse?		Query enabled/disabled 4-wire sense state

Description :RSENs enables or disables 4-wire sensing for RTD and thermistor temperature transducers. 4-wire sensing should be used for RTD and thermistor sensors with <1kΩ resistance.

Ground connect mode

:GCONnect <b>

:SYSTem:GCONnect <b> — Control ground connect mode

Parameters <b> = 1 or ON — Ground connect enabled; 0 or OFF — Ground connect disabled

Query :GCONnect? — Query ground connect mode

Description This command controls the ground connect mode. With the ground connect mode enabled, the INPUT and OUTPUT (-) terminals are all internally connected to analog common. See Section 2, "Connections" for details. The front panel "REAR" annunciator turns on when ground connect is enabled.

RS-232 interface

NOTE *Although these commands are intended for use over the RS-232, they also function with the GPIB.*

:LOCal

:SYSTem:LOCal — Take Model 2510 out of remote

Description Normally, during RS-232 communications, front panel keys are operational. However, the user may wish to lock out front panel keys during RS-232 communications. See :RWLock.

This command is used to remove the Model 2510 from the remote state and enables the operation of front panel keys. Its operation is similar to the IEEE-488 GTL command (Section 6).

:REMote

:SYSTem:REMote — Place the Model 2510 in remote

Description This command is used to place the Model 2510 in the remote state. In remote, the front panel keys will be locked out if local lockout is asserted. See :RWLock. :REMote performs the same function as the IEEE-488 REN command (Section 6).

:RWLock

:SYSTem:RWLock — Disable front panel keys

Description This command is used to enable or disable local lockout. When enabled, the front panel keys are locked out (not operational) when the instrument is in remote. (See :REMote.) When disabled, the front panel keys are operational in remote. :RWLock performs the same function as the IEEE-488 LLO command (Section 6).

NOTE *OUTPUT OFF is still operational in local lockout.*

Trigger subsystem

The Trigger subsystem is made up of commands to configure triggering. These commands are summarized in Table 10-9.

NOTE *The trigger commands function only in the calibration mode, and they are ignored during normal operation. See the Model 2510 Service Manual for details on calibrating the unit.*

Initiate control/measure cycle

:INITiate

:INITiate[:IMMediate] Take Model 2510 out of idle state

Description This command is used to initiate control-measure operation by taking the Model 2510 out of idle. The :READ? and :MEASure? commands also perform an initiation.

Abort source/measure cycle

:ABORt

Abort operation

Description When this action command is sent, the Model 2510 aborts operation, returns to the idle state, and turns the output off.

UNIT subsystem

The UNIT subsystem controls units used for the temperature function. These commands are summarized in Table 10-10.

:TEMPerature <name>

:UNIT:TEMPerature <name> Set temperature units

Parameters	<name> =	C | CEL	Celsius units
		F | FAR	Fahrenheit units
		K	Kelvin units

Query :TEMPerature? Query temperature units

Description This command sets the units for the temperature function. Temperature units include °C, °F, and K. Note that both temperature command parameter units and reading temperature units are affected by this command.

A
Specifications

The Model 2510 Thermoelectric Cooler Controller is designed to:

- control the power to the TEC to maintain a constant temperature, current, voltage, or thermistor resistance
- measure the resistance of the TEC
- software PID loop

CONTROL SYSTEM SPECIFICATIONS

SET: Constant Peltier Temperature
Constant Peltier Voltage
Constant Peltier Current
Constant Thermistor Resistance

CONTROL METHOD:

Programmable software PID loop.

Proportional, Integral, and Derivative gains independently programmable.

SETPOINT SHORT TERM STABILITY: ±0.005°C rms.[2, 3]

SETPOINT LONG TERM STABILITY: ±0.01°C.[2, 4]

SETPOINT RANGE: –50°C to 225°C.

OVER TEMPERATURE LIMIT: 250°C max.

UNDER TEMPERATURE LIMIT: –50°C max.

SETPOINT RESOLUTION: 0.001°C, 1mV, 100µA, 0.01% of nominal (25°C) thermistor resistance.

HARDWARE CURRENT LIMIT: 1.0A to 5.25A ±5%.

SOFTWARE VOLTAGE LIMIT:±0.5 to 10.5V ±5%.

TEC OUTPUT SPECIFICATIONS

OUTPUT RANGE: ±10 VDC at up to ±5 ADC.

OUTPUT RIPPLE: <5mV rms.[5]

AC RESISTANCE EXCITATION: ±(9.6mA + 190µA).[10, 11]

TEC MEASUREMENT SPECIFICATIONS

FUNCTION	1 Year, 23°C ±5°C
Operating Resistance [1, 6, 7, 8]	±(2.0% of rdg + 0.1Ω)
Operating Voltage [1, 8]	±(0.1% of rdg + 4mV)
Operating Current [8]	±(0.4% of rdg + 8mA)
AC Resistance [1, 14]	±(0.10% of rdg + 0.02Ω)

OPEN SHORTED THERMOELECTRIC DETECTION

LOAD IMPEDANCE: Stable into 1µF typical.

COMMON MODE VOLTAGE: 30VDC maximum.

COMMON MODE ISOLATION: >10^9Ω, <1500pF.

MAX. SENSE LEAD RESISTANCE: 1Ω for rated accuracy.

MAX. FORCE LEAD RESISTANCE: 0.1Ω.

THERMAL FEEDBACK ELEMENT SPECIFICATIONS (1 Year, 23°C ±5°C)

Sensor Type	RTD	Thermistor				Solid State	
	100 Ω	100 Ω	1 kΩ	10 kΩ	100 kΩ	Current Output (Iss)	Voltage Output (Vss)
Excitation [9]	2.5 mA	2.5 mA	833 µA	100 µA	33 µA	+13.5 V	2.5 mA
Compliance						833 µA max	15.75 V max
Nominal Resistance Range	0–250 Ω	0–1 kΩ	0–10 kΩ	0–80 kΩ	0–200 kΩ		
Excitation Accuracy	±2.9%	±2.9%	±2.9%	±2.9%	±2.9%	±12%	±2.9%
Nominal Sensor Temperature Range	–50° to +250°C	–50° to +250°C	–50° to +250°C	–50° to +250°C	–50° to +250°C	–40° to +100°C	–40° to +100°C
Sensor Coefficients	α, β, δ	A, B, C	A, B, C	A, B, C	A, B, C	Slope & offset	Slope & offset
Measurement Accuracy ±(% rdg + offset)	0.04 + 0.07 Ω[1]	0.04 + 0.07 Ω[1]	0.04 + 0.4 Ω[1]	0.02 + 3 Ω[1]	0.04 + 21 Ω	0.03 + 100 nA	0.03 + 500 µV

THERMISTOR MEASUREMENT ACCURACY[15]

Nominal Thermistor Resistance	Accuracy vs. Temperature 0°C	25°C	50°C	100°C
100 Ω	0.021°C	0.035°C	0.070°C	0.27°C
1 kΩ	0.015°C	0.023°C	0.045°C	0.18°C
10 kΩ	0.006°C	0.012°C	0.026°C	0.15°C
100 kΩ	0.009°C	0.014°C	0.026°C	0.13°C

OPEN/SHORTED ELEMENT DETECTION

SOFTWARE LINEARIZATION FOR THERMISTOR AND RTD

COMMON MODE VOLTAGE: 30VDC.

COMMON MODE ISOLATION: >10^9Ω, <1000pF

MAX. VOLTAGE DROP IN INPUT FORCE LEADS: 1 volt.

MAX. SENSE LEAD RESISTANCE: 100Ω for rated accuracy.

SENSE INPUT IMPEDANCE: > $1 \cdot 10^8$Ω.

GENERAL

NOISE REJECTION:

SPEED	NPLC	CMRR[13]
Normal	1.00	90 dB

SOURCE OUTPUT MODES: Fixed DC level.

PROGRAMMABILITY: IEEE-488 (SCPI-1995.0), RS-232, 3 user-definable power-up states plus factory default and *RST.

POWER SUPPLY: Nominal 100 to 240VAC rms, 50–60Hz, 90VA.

WARRANTY: 1 year.

EMC: Conforms to European Union Directive 89/336/EEC, EN 61326-1.

SAFETY: Conforms to European Union Directive 73/23/EEC, EN 61010-1.

VIBRATION: MIL-PRF-28800F Class 3 Random Vibration.

WARM-UP: 1 hour to rated accuracies.

DIMENSIONS, WEIGHT: 89mm high × 213 mm wide × 370mm deep (3½ in × 8⅜ in × 14⅝ in). **Bench configuration (with handle & feet):** 104mm high × 238mm wide × 370mm deep (4⅛ in × 9⅜ in × 14⅝ in). **Net Weight:** 3.8kg (8.38 lbs).

ENVIRONMENT: Operating: 0°–50°C, 70% R.H. up to 35°C. Derate 3% R.H./°C, 35°–50°C. **Storage:** –25° to 65°C

NOTES

1 With remote voltage sense.
2 With 10kΩ thermistor as sensor.
3 Short term stability is defined as 24 hours with Peltier and Model 2510 at 25°C ±0.5°C.
4 Long term stability is defined as 30 days with Peltier and Model 2510 at 25°C ±0.5°C.
5 10Hz to 10MHz measured at 5A output into a 2Ω load.
6 Common mode voltage = 0V (meter connect enabled, connects Peltier low output to thermistor measure circuit ground). ±(0.1% of rdg + 0.1Ω) with meter connect disabled.
7 Resistance range 0Ω to 20Ω for rated accuracy.
8 Current through Peltier > 0.2A.
9 Default values shown, selectable values of 3μA, 10μA, 33μA, 100μA, 833μA, 2.5mA. Note that temperature control performance will degrade at lower currents.
10 AC Ohms is a dual pulsed measurement using current reversals available over bus only.
11 @23°C ±5°C.
12 For line frequency ±0.1%.
13 For 1kΩ unbalance in LO lead. Minimum amplifier specification.
14 Resistance range 0Ω to 100Ω for rated accuracy.
15 Accuracy figures are for typical thermistors at the Model 2510 default thermistor current. Based on ITS-90.

Specifications subject to change without notice.

B
Status and Error Messages

Introduction

This Appendix contains a summary of status and error messages, which status register bits are set when messages occur, and methods to avoid or eliminate most common SCPI errors.

Status and error messages

Table B-1 summarizes status and error messages, which are stored in the Error Queue. Each message is preceded by a code number. Negative (-) numbers are used for SCPI-defined messages, and positive (+) numbers are used for Keithley-defined messages. Note that error and status conditions will also set specific bits in various status registers, as summarized in Table B-1.

Section 7 has detailed information on registers and queues. Briefly, you can use the following queries to obtain error and status information:

- :SYST:ERR? — reads Error Queue.
- *ESR? — reads Standard Event Status Register.
- :STAT:OPER? — reads Operation Event Register.
- :STAT:MEAS? — reads Measurement Event Register.
- :STAT:QUES? — reads Questionable Event Register.

NOTE *SCPI-confirmed messages are described in volume 2: Command Reference of the Standard Commands for Programmable Instruments. Refer to the :SYSTem:ERRor? command.*

Table B-1
Status and error messages

Number	Error message	Event[1]	Status register[2]	Bit
-440	Query UNTERMINATED after indefinite response	EE	Standard Event	2
-430	Query DEADLOCKED	EE	Standard Event	2
-420	Query UNTERMINATED	EE	Standard Event	2
-410	Query INTERRUPTED	EE	Standard Event	2
-363	Input buffer overrun	EE	Standard Event	3
-362	Framing error in program message	EE	Standard Event	3
-361	Parity error in program message	EE	Standard Event	3
-360	Communications error	EE	Standard Event	3
-350	Queue overflow	SYS	Standard Event	3
-330	Self-test failed	EE	Standard Event	3
-314	Save/recall memory lost	EE	Standard Event	3
-315	Configuration memory lost	EE	Standard Event	3
-285	Program syntax error	EE	Standard Event	4
-284	Program currently running	EE	Standard Event	4
-282	Illegal program name	EE	Standard Event	4
-281	Cannot create program	EE	Standard Event	4
-260	Expression error	EE	Standard Event	4
-241	Hardware missing	EE	Standard Event	4
-230	Data corrupt or stale	EE	Standard Event	4
-225	Out of memory	EE	Standard Event	4
-224	Illegal parameter value	EE	Standard Event	4
-223	Too much data	EE	Standard Event	4
-222	Parameter data out of range	EE	Standard Event	4
-221	Settings conflict	EE	Standard Event	4
-220	Parameter error	EE	Standard Event	4
-215	Arm deadlock	EE	Standard Event	4
-214	Trigger deadlock	EE	Standard Event	4
-213	Init ignored	EE	Standard Event	4
-212	Arm ignored	EE	Standard Event	4
-211	Trigger ignored	EE	Standard Event	4
-210	Trigger error	EE	Standard Event	4
-202	Settings lost due to rtl	EE	Standard Event	4
-201	Invalid while in local	EE	Standard Event	4
-200	Execution error	EE	Standard Event	4

Table B-1 (cont.)
Status and error messages

Number	Error message	Event[1]	Status register[2]	Bit
-178	Expression data not allowed	EE	Standard Event	5
-171	Invalid expression	EE	Standard Event	5
-170	Expression error	EE	Standard Event	5
-168	Block data not allowed	EE	Standard Event	5
-161	Invalid block data	EE	Standard Event	5
-160	Block data error	EE	Standard Event	5
-158	String data not allowed	EE	Standard Event	5
-154	String too long	EE	Standard Event	5
-151	Invalid string data	EE	Standard Event	5
-150	String data error	EE	Standard Event	5
-148	Character data not allowed	EE	Standard Event	5
-144	Character data too long	EE	Standard Event	5
-141	Invalid character data	EE	Standard Event	5
-140	Character data error	EE	Standard Event	5
-128	Numeric data not allowed	EE	Standard Event	5
-124	Too many digits	EE	Standard Event	5
-123	Exponent too large	EE	Standard Event	5
-121	Invalid character in number	EE	Standard Event	5
-120	Numeric data error	EE	Standard Event	5
-114	Header suffix out of range	EE	Standard Event	5
-113	Undefined header	EE	Standard Event	5
-112	Program mnemonic too long	EE	Standard Event	5
-111	Header separator error	EE	Standard Event	5
-110	Command header error	EE	Standard Event	5
-109	Missing parameter	EE	Standard Event	5
-108	Parameter not allowed	EE	Standard Event	5
-105	GET not allowed	EE	Standard Event	5
-104	Data type error	EE	Standard Event	5
-103	Invalid separator	EE	Standard Event	5
-102	Syntax error	EE	Standard Event	5
-101	Invalid character	EE	Standard Event	5
-100	Command error	EE	Standard Event	5
+000	No error	SE		

Table B-1 (cont.)
Status and error messages

Number	Error message	Event[1]	Status register[2]	Bit
	Measurement events:			
+100	Over Temperature	SE	Measurement Event	0
+101	Under Temperature	SE	Measurement Event	1
+102	Over Voltage	SE	Measurement Event	2
+103	Current Overload	SE	Measurement Event	3
+104	Over Resistance	SE	Measurement Event	4
+105	Under Resistance	SE	Measurement Event	5
+106	Reading available	SE	Measurement Event	6
+107	Reading overflow	SE	Measurement Event	7
+111	OUTPUT Enable asserted	SE	Measurement Event	11
	Standard events:			
+200	Operation complete	SE	Standard Event	0
	Operation events:			
+300	Device calibrating	SE	Operation Event	0
+303	Device sweeping	SE	Operation Event	3
+305	Waiting in trigger layer	SE	Operation Event	5
+306	Waiting in arm layer	SE	Operation Event	6
+310	Entering idle layer	SE	Operation Event	10
	Questionable events:			
+408	Questionable Calibration	SE	Questionable Event	8
+414	Command Warning	SE	Questionable Event	14
	Calibration errors:			
+500	Date of calibration not set	EE	Standard Event	3
+501	Next date of calibration not set	EE	Standard Event	3
+502	Calibration data invalid	EE	Standard Event	3
+505	Source offset data invalid	EE	Standard Event	3
+506	Source gain data invalid	EE	Standard Event	3
+507	Measurement offset data invalid	EE	Standard Event	3
+508	Measurement gain data invalid	EE	Standard Event	3
+509	Not permitted with cal locked	EE	Standard Event	3
+510	Not permitted with cal un-locked	EE	Standard Event	3

Table B-1 (cont.)
Status and error messages

Number	Error message	Event[1]	Status register[2]	Bit
	Lost data errors:			
+602	GPIB address lost	EE	Standard Event	3
+603	Power-on state lost	EE	Standard Event	3
+604	DC calibration data lost	EE	Standard Event	3
+605	Calibration dates lost	EE	Standard Event	3
+606	GPIB communication language lost	EE	Standard Event	3
	Communication errors:			
+700	Invalid system communication	EE	Standard Event	3
+701	ASCII only with RS-232	EE	Standard Event	3
	Additional command execution errors:			
+800	Illegal with storage active	EE	Standard Event	4
+801	Not permitted with OUTPUT on	EE	Standard Event	4
+802	OUTPUT blocked by OUTPUT enable	EE	Standard Event	4
+803	Not permitted with OUTPUT off	EE	Standard Event	4
+804	OUTPUT blocked by Over Temp	EE	Standard Event	4
+805	OUTPUT blocked by Under Temp	EE	Standard Event	4
+806	Invalid while taking reading	EE	Standard Event	4
+900	Internal System Error	EE	Standard Event	4

1. EE = Error Event
   SE = Status Event
   SYS = System Error Event
2. Use following queries to read status registers:
   Standard Event: *ESR?
   Operation Event: STAT:OPER?
   Measurement Event: STAT:MEAS?
   Questionable Event: STAT:QUES?

Eliminating common SCPI errors

There are three SCPI errors that occur more often than any others:

- -113, "Undefined header"
- -410, "Query INTERRUPTED"
- -420, "Query UNTERMINATED"

The following paragraphs discuss the most common causes for these errors and methods for avoiding them.

-113, "Undefined header"

This error indicates that the command you sent to the instrument did not contain a recognizable command name. The most likely causes for this error are:

- *Missing space between the command and its parameter.* There must be one or more spaces (blanks) between the command and its parameter. For example:

:SENS:TEMP:THER:RANGl00	Incorrect (no space between command and parameter)
:SENS:TEMP:THER:RANG 100	Correct

- *Improper short or long form.* Check the command lists in Section 10 for the correct command name.
- *Blanks (spaces) within the command name.* For example:

:SYST :ERR?	Incorrect (space between :SYST and :ERR?)
:SYST:ERR?	Correct

-410, "Query INTERRUPTED"

This error occurs when you have sent a valid query to the instrument, and then send it another command or query, or a Group Execute Trigger (GET) before it has had a chance to send the entire response message (including the line-feed/EOI terminator). The most likely causes are:

- *Sending a query to the instrument and then sending another command or query before reading the response to the first query.* For example, the following sequence of commands will cause an error -410:

  :SYST:ERR?

  *OPC?

  This sequence generates an error because you must read the response to :SYST:ERR? before sending the *OPC? query.
- *Incorrectly configured IEEE-488 driver.* The driver must be configured so that when talking on the bus it sends line-feed with EOI as the terminator, and when listening on the bus it expects line-feed with EOI as the terminator. See the reference manual for your particular IEEE-488 interface.

- *Errors when transferring binary data.* Many programs such as Test Point, Lab View, or Visual Basic default to terminating communications with a carriage return (CR) and/or line feed (LF). When transmitting binary data, a random bit pattern could be misinterpreted as a CR or LF and terminate communication before all bytes are transmitted, resulting in this error. To avoid this problem with binary transfers, change the default terminator to none to avoid problems, and terminate on EOI only. See the :FORM:DATA command in "FORMat subsystem" in Section 10 for information on selecting the binary transfer data format.

-420, "Query UNTERMINATED"

This error occurs when you address the instrument to talk, and there is no response message to send. The most likely causes are:

- *Not sending a query.* You must send a valid query to the instrument before addressing it to talk.
- *Sending an invalid query.* If you have sent a query and still get this error, make sure that the instrument is processing the query without error. For example, sending an ill-formed query that generates an error -113, "Undefined header" and then addressing the instrument to talk will generate an error -420, "Query UNTERMINATED" as well.
- *Valid query following an invalid command.* This situation can occur when you send multiple commands or queries (program message units) within one command string (program message). When the Model 2510 detects an error in a program message unit, it discards all further program message units until the end of the string; for example:

  :SENS:VOLT?; :SOUR:FUNC?

  In the above program message, the program message unit :SENS:VOLT? will generate error -113, "Undefined header," and the Model 2510 will discard the second program message unit :SOUR:FUNC? even though it is a valid query.

C
IEEE-488 Bus Overview

Introduction

Basically, the IEEE-488 bus is a communication system between two or more electronic devices. A device can be either an instrument or a computer. When a computer is used on the bus, it serves to supervise the communication exchange between all the devices and is known as the controller. Supervision by the controller consists of determining which device will talk and which device will listen. As a talker, a device will output information and as a listener, a device will receive information. To simplify the task of keeping track of the devices, a unique address number is assigned to each one.

On the bus, only one device can talk at a time and is addressed to talk by the controller. The device that is talking is known as the active talker. The devices that need to listen to the talker are addressed to listen by the controller. Each listener is then referred to as an active listener. Devices that do not need to listen are instructed to unlisten. The reason for the unlisten instruction is to optimize the speed of bus information transfer since the task of listening takes up bus time.

Through the use of control lines, a handshake sequence takes place in the transfer process of information from a talker to a listener. This handshake sequence helps ensure the credibility of the information transfer. The basic handshake sequence between an active controller (talker) and a listener is as follows:

1. The listener indicates that it is ready to listen.
2. The talker places the byte of data on the bus and indicates that the data is available to the listener.
3. The listener, aware that the data is available, accepts the data and then indicates that the data has been accepted.
4. The talker, aware that the data has been accepted, stops sending data and indicates that data is not being sent.
5. The listener, aware that there is no data on the bus, indicates that it is ready for the next byte of data.

Bus description

The IEEE-488 bus, which is also frequently referred to as the GPIB (General Purpose Interface Bus), was designed as a parallel transfer medium to optimize data transfer without using an excessive number of bus lines. In keeping with this goal, the bus has only eight data lines that are used for both data and with most commands. Five bus management lines and three handshake lines round out the complement of bus signal lines.

A typical set up for controlled operation is shown in Figure C-1. Generally, a system will contain one controller and a number of other instruments to which the commands are given. Device operation is categorized into three operators: controller, talker, and listener. The controller does what its name implies; it controls the instruments on the bus. The talker sends data while a listener receives data. Depending on the type of instrument, any particular device can be a talker only, a listener only, or both a talker and listener.

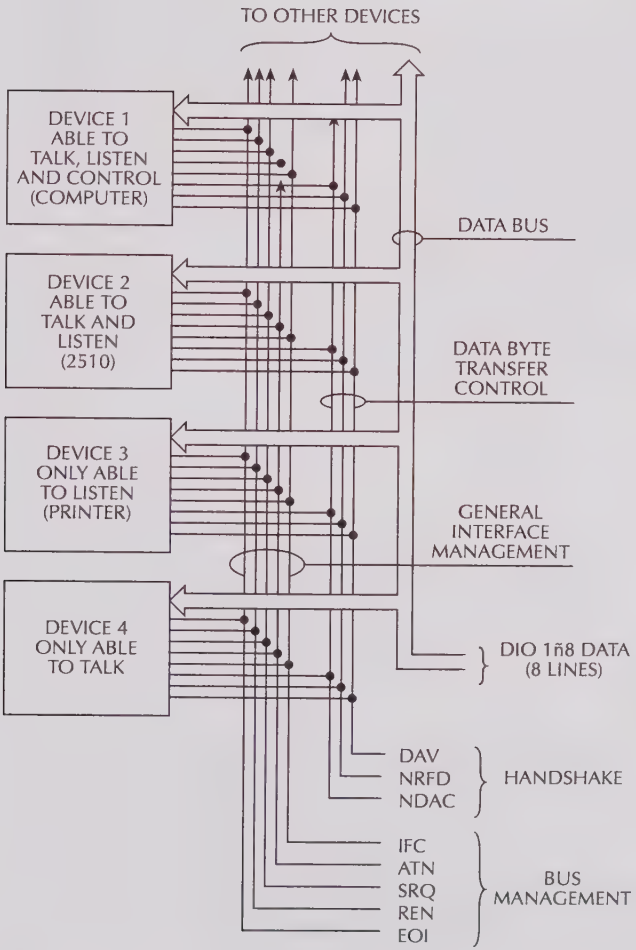


Figure C-1
IEEE-488 bus configuration

There are two categories of controllers: system controller and basic controller. Both are able to control other instruments, but only the system controller has the absolute authority in the system. In a system with more than one controller, only one controller may be active at any given time. Certain protocol is used to pass control from one controller to another.

The IEEE-488 bus is limited to 15 devices, including the controller. Thus, any number of talkers and listeners up to that limit may be present on the bus at one time. Although several devices may be commanded to listen simultaneously, the bus can have only one active talker, or communications would be scrambled.

A device is placed in the talk or listen state by sending an appropriate talk or listen command. These talk and listen commands are derived from an instrument's primary address. The primary address may have any value between 0 and 31, and is generally set by rear panel DIP

switches or programmed from the front panel of the instrument. The actual listen address value sent out over the bus is obtained by ORing the primary address with #H20. For example, if the primary address is #H16, the actual listen address is #H36 (#H36 = #H16 + #H20). In a similar manner, the talk address is obtained by ORing the primary address with #H40. With the present example, the talk address derived from a primary address of 16 decimal would be #H56 (#H56 = #H16 + #H40).

The IEEE-488 standards also include another addressing mode called secondary addressing. Secondary addresses lie in the range of #H60-#H7F. Note, however, that many devices, including the Model 2510, do not use secondary addressing.

Once a device is addressed to talk or listen, the appropriate bus transactions take place. For example: if the instrument is addressed to talk, it places its data string on the bus one byte at a time. The controller reads the information and the appropriate software can be used to direct the information to the desired location.

Bus lines

The signal lines on the IEEE-488 bus are grouped into three different categories: data lines, management lines, and handshake lines. The data lines handle bus data and commands, while the management and handshake lines ensure that proper data transfer and operation takes place. Each bus line is active low, with approximately zero volts representing a logic 1(true). The following paragraphs describe the operation of these lines.

Data lines

The IEEE-488 bus uses eight data lines that transfer data one byte at a time. DIO1 (Data Input/Output) through DIO8 (Data Input/Output) are the eight data lines used to transmit both data and multiline commands and are bidirectional. The data lines operate with low true logic.

Bus management lines

The five bus management lines help to ensure proper interface control and management. These lines are used to send the uniline commands.

ATN (Attention) — The ATN line is one of the more important management lines. The state of this line determines how information on the data bus is to be interpreted.

IFC (Interface Clear) — As the name implies, the IFC line controls clearing of instruments from the bus.

REN (Remote Enable) — The REN line is used to place the instrument on the bus in the remote mode.

EOI (End or Identify) — The EOI is usually used to mark the end of a multi-byte data transfer sequence.

SRQ (Service Request) — This line is used by devices when they require service from the controller.

Handshake lines

The bus handshake lines operate in an interlocked sequence. This method ensures reliable data transmission regardless of the transfer rate. Generally, data transfer will occur at a rate determined by the slowest active device on the bus.

One of the three handshake lines is controlled by the source (the talker sending information), while the remaining two lines are controlled by accepting devices (the listener or listeners receiving the information). The three handshake lines are:

DAV (DATA VALID) — The source controls the state of the DAV line to indicate to any listening devices whether or not data bus information is valid.

NRFD (Not Ready For Data) — The acceptor controls the state of NRFD. It is used to signal to the transmitting device to hold off the byte transfer sequence until the accepting device is ready.

NDAC (Not Data Accepted) — NDAC is also controlled by the accepting device. The state of NDAC tells the source whether or not the device has accepted the data byte.

The complete handshake sequence for one data byte is shown in Figure C-2. Once data is placed on the data lines, the source checks to see that NRFD is high, indicating that all active devices are ready. At the same time, NDAC should be low from the previous byte transfer. If these conditions are not met, the source must wait until NDAC and NRFD have the correct status. If the source is a controller, NRFD and NDAC must be stable for at least 100nsec after ATN is set true. Because of the possibility of a bus hang up, many controllers have time-out routines that display messages in case the transfer sequence stops for any reason.

Once all NDAC and NRFD are properly set, the source sets DAV low, indicating to accepting devices that the byte on the data lines is now valid. NRFD will then go low, and NDAC will go high once all devices have accepted the data. Each device will release NDAC at its own rate, but NDAC will not be released to go high until all devices have accepted the data byte.

The sequence just described is used to transfer both data, talk and listen addresses, as well as multiline commands. The state of the ATN line determines whether the data bus contains data, addresses or commands as described in the following paragraph.

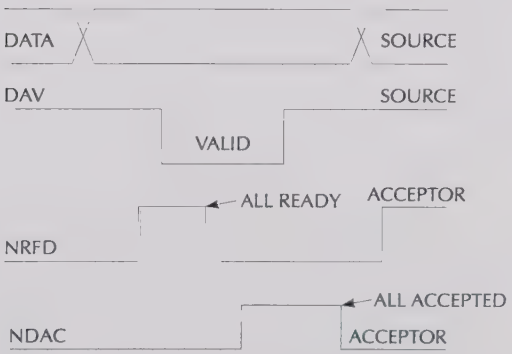


Figure C-2 IEEE-488 handshake sequence

Bus commands

The instrument may be given a number of special bus commands through the IEEE-488 interface. This section briefly describes the purpose of the bus commands which are grouped into the following four categories.

1. **Uniline commands** — Sent by setting the associated bus lines true. For example, to assert REN (Remote Enable), the REN line would be set low (true).
2. **Multiline commands** — General bus commands which are sent over the data lines with the ATN line true (low).
3. **Common commands** — Commands that are common to all devices on the bus; sent with ATN high (false).
4. **SCPI commands** — Commands that are particular to each device on the bus; sent with ATN (false).

These bus commands and their general purpose are summarized in Table C-1.

Table C-1
IEEE-488 bus command summary

Command type	Command	State of ATN line	Comments
Uniline	REN (Remote Enable)	X	Set up devices for remote operation.
	EOI (End Or Identify)	X	Marks end of transmission.
	IFC (Interface Clear)	X	Clears interface.
	ATN (Attention)	Low	Defines data bus contents.
	SRQ (Service Request)	X	Controlled by external device.
Multiline Universal	LLO (Local Lockout)	Low	Locks out local operation.
	DCL (Device Clear)	Low	Returns device to default conditions.
	SPE (Serial Poll Enable)	Low	Enables serial polling.
	SPD (Serial Poll Disable)	Low	Disables serial polling.
Addressed	SDC (Selective Device Clear)	Low	Returns unit to default conditions.
	GTL (Go To Local)	Low	Returns device to local.
Unaddressed	UNL (Unlisten)	Low	Removes all listeners from the bus.
	UNT (Untalk)	Low	Removes any talkers from the bus.
Common	–	High	Programs IEEE-488.2 compatible instruments for common operations.
SCPI	–	High	Programs SCPI compatible instruments for particular operations.

Uniline commands

ATN, IFC and REN are asserted only by the controller. SRQ is asserted by an external device. EOI may be asserted either by the controller or other devices depending on the direction of data transfer. The following is a description of each command. Each command is sent by setting the corresponding bus line true.

REN (Remote Enable) — REN is sent to set up instruments on the bus for remote operation. When REN is true, devices will be removed from the local mode. Depending on device configuration, all front panel controls except the LOCAL key (if the device is so equipped) may be locked out when REN is true. Generally, REN should be sent before attempting to program instruments over the bus.

EOI (End or Identify) — EOI is used to positively identify the last byte in a multi-byte transfer sequence, thus allowing data words of various lengths to be transmitted easily.

IFC (Interface Clear) — IFC is used to clear the interface and return all devices to the talker and listener idle states.

ATN (Attention) — The controller sends ATN while transmitting addresses or multiline commands.

SRQ (Service Request) — SRQ is asserted by a device when it requires service from a controller.

Universal multiline commands

Universal commands are those multiline commands that require no addressing. All devices equipped to implement such commands will do so simultaneously when the commands are transmitted. As with all multiline commands, these commands are transmitted with ATN true.

LLO (Local Lockout) — LLO is sent to the instrument to lock out the LOCAL key and thus all the front panel controls.

DCL (Device Clear) — DCL is used to return instruments to some default state. Usually, instruments return to the power-up conditions.

SPE (Serial Poll Enable) — SPE is the first step in the serial polling sequences, which is used to determine which device has requested service.

SPD (Serial Poll Disable) — SPD is used by the controller to remove all devices on the bus from the serial poll mode and is generally the last command in the serial polling sequence.

Addressed multiline commands

Addressed commands are multiline commands that must be preceded by the device listen address before that instrument will respond to the command in question. Note that only the addressed device will respond to these commands. Both the commands and the address preceding it are sent with ATN true.

SDC (Selective Device Clear) — The SDC command performs essentially the same function as the DCL command except that only the addressed device responds. Generally, instruments return to their power-up default conditions when responding to the SDC command.

GTL (Go To Local) — The GTL command is used to remove instruments from the remote mode. With some instruments, GTL also unlocks front panel controls if they were previously locked out with the LLO command.

GET (Group Execute Trigger) — The GET command is used to trigger devices to perform a specific action that depends on device configuration (for example, take a reading). Although GET is an addressed command, many devices respond to GET without addressing.

Address commands

Addressed commands include two primary command groups and a secondary address group. ATN is true when these commands are asserted. The commands include:

LAG (Listen Address Group) — These listen commands are derived from an instrument's primary address and are used to address devices to listen. The actual command byte is obtained by ORing the primary address with #H20.

TAG (Talk Address Group) — The talk commands are derived from the primary address by ORing the address with #H40. Talk commands are used to address devices to talk.

SCG (Secondary Command Group) — Commands in this group provide additional addressing capabilities. Many devices (including the Model 2510) do not use these commands.

Unaddress commands

The two unaddress commands are used by the controller to remove any talkers or listeners from the bus. ATN is true when these commands are asserted.

UNL (Unlisten) — Listeners are placed in the listener idle state by the UNL command.

UNT (Untalk) — Any previously commanded talkers will be placed in the talker idle state by the UNT command.

Common commands

Common commands are commands that are common to all devices on the bus. These commands are designated and defined by the IEEE-488.2 standard.

Generally, these commands are sent as one or more ASCII characters that tell the device to perform a common operation, such as reset. The IEEE-488 bus treats these commands as data in that ATN is false when the commands are transmitted.

SCPI commands

SCPI commands are commands that are particular to each device on the bus. These commands are designated by the instrument manufacturer, and are based on the instrument model defined by the Standard Commands for Programmable Instruments (SCPI) Consortium's SCPI standard.

Generally, these commands are sent as one or more ASCII characters that tell the device to perform a particular operation, such as setting a range or closing a relay. The IEEE-488 bus treats these commands as data in that ATN is false when the commands are transmitted.

Command codes

Command codes for the various commands that use the data lines are summarized in Figure C-3. Hexadecimal and the decimal values for the various commands are listed in Table C-2.

Table C-2
Hexadecimal and decimal command codes

Command	Hex value	Decimal value
GTL	01	1
SDC	04	4
GET	08	8
LLO	11	17
DCL	14	20
SPE	18	24
SPD	19	25
LAG	20-3F	32-63
TAG	40-5F	64-95
SCG	60-7F	96-127
UNL	3F	63
UNT	5F	95

Figure C-3
Command codes

D_7 D_6 D_5 D_4					X 0 0 0	Command	X 0 0 1	Command	X 0 1 0	Primary Address	X 0 1 1	Primary Address	X 1 0 0	Primary Address	X 1 0 1	Primary Address	X 1 1 0		X 1 1 1	
Bits	D_3 →	D_2 →	D_1 →	D_0 →	Column → Row →	0 (A)	0 (B)	1 (A)	1 (B)	2 (A)	2 (B)	3 (A)	3 (B)	4 (A)	4 (B)	5 (A)	5 (B)	6 (A)	6 (B)	7 (A)	7 (B)
	0	0	0	0	0	NUL		DIF		SP	0	0	16	@	0	P	16			p	
	0	0	0	1	1	SOH	GTL	DC1	LLO	!	1	1	17	A	1	Q	17	a		q	
	0	0	1	0	2	STX		DC2		"	2	2	18	B	2	R	18	b		r	
	0	0	1	1	3	ETX		DC3		#	3	3	19	C	3	S	19	c		s	
	0	1	0	0	4	EOT	SDC	DC4	DCL	$	4	4	20	D	4	T	20	d		t	
	0	1	0	1	5	ENQ	PPC*	NAK	PPU*	%	5	5	21	E	5	U	21	e		u	
	0	1	1	0	6	ACK		SYN		&	6	6	22	F	6	V	22	f		v	
	0	1	1	1	7	BEL		ETB		'	7	7	23	G	7	W	23	g		w	
	1	0	0	0	8	BS	GET	CAN	SPE	(	8	8	24	H	8	X	24	h		x	
	1	0	0	1	9	HT	TCT*	EM	SPD	)	9	9	25	I	9	Y	25	i		y	
	1	0	1	0	10	LF		SUB		•	10	:	26	J	10	Z	26	j		z	
	1	0	1	1	11	VT		ESC		+	11	;	27	K	11	[	27	k		{	
	1	1	0	0	12	FF		FS		,	12	<	28	L	12	\	28	l		:	
	1	1	0	1	13	CR		GS		–	13	=	29	M	13	]	29	m		}	
	1	1	1	0	14	SO		RS		.	14	>	30	N	14	^	30	n		≅	
	1	1	1	1	15	SI		US		/	15	?	UNL	O	15	—	UNT	o		DEL	

ADDRESSED COMMAND GROUP (ACG)

UNIVERSAL COMMAND GROUP (UCG)

LISTEN ADDRESS GROUP (LAG)

TALK ADDRESS GROUP (TAG)

PRIMARY COMMAND GROUP (PCG)

SECONDARY COMMAND GROUP (SDC)

*PPC (PARALLEL POLL CONFIGURE) PPU (PARALLEL POLL UNCONFIGURE), and TCT (TAKE CONTROL) not implemented by Model 2510.
Note: D_0 = D101 ...D_7 = D108; X = Don't Care.

Typical command sequences

For the various multiline commands, a specific bus sequence must take place to properly send the command. In particular, the correct listen address must be sent to the instrument before it will respond to addressed commands. Table C-3 lists a typical bus sequence for sending the addressed multiline commands. In this instance, the SDC command is being sent to the instrument. UNL is generally sent as part of the sequence to ensure that no other active listeners are present. Note that ATN is true for both the listen command and the SDC command byte itself.

Table C-3
Typical addressed multiline command sequence

Step	Command	ATN state	Data bus		
			ASCII	Hex	Decimal
1	UNL	Set low	?	3F	63
2	LAG*	Stays low	8	38	56
3	SDC	Stays low	EOT	04	4
4		Returns high			

*Assumes primary address = 24.

Table C-4 gives a typical common command sequence. In this instance, ATN is true while the instrument is being addressed, but it is set high while sending the common command string.

Table C-4
Typical addressed common command sequence

Step	Command	ATN state	Data bus		
			ASCII	Hex	Decimal
1	UNL	Set low	?	3F	63
2	LAG*	Stays low	8	38	56
3	Data	Set high	*	2A	42
4	Data	Stays high	R	52	82
5	Data	Stays high	S	53	83
6	Data	Stays high	T	54	84

*Assumes primary address = 24.

IEEE command groups

Command groups supported by the Model 2510 are listed in Table C-5. Common commands and SCPI commands are not included in this list.

Table C-5
IEEE command groups

HANDSHAKE COMMAND GROUP	
	NDAC = NOT DATA ACCEPTED
	NRFD = NOT READY FOR DATA
	DAV = DATA VALID
UNIVERSAL COMMAND GROUP	
	ATN = ATTENTION
	DCL = DEVICE CLEAR
	IFC = INTERFACE CLEAR
	REN = REMOTE ENABLE
	SPD = SERIAL POLL DISABLE
	SPE = SERIAL POLL ENABLE
ADDRESS COMMAND GROUP	
LISTEN	LAG = LISTEN ADDRESS GROUP
	MLA = MY LISTEN ADDRESS
	UNL = UNLISTEN
TALK	TAG = TALK ADDRESS GROUP
	MTA = MY TALK ADDRESS
	UNT = UNTALK
	OTA = OTHER TALK ADDRESS
ADDRESSED COMMAND GROUP	
	ACG = ADDRESSED COMMAND GROUP
	GTL = GO TO LOCAL
	SDC = SELECTIVE DEVICE CLEAR
STATUS COMMAND GROUP	
	RQS = REQUEST SERVICE
	SRQ = SERIAL POLL REQUEST
	STB = STATUS BYTE
	EOI = END

Interface function codes

The interface function codes, which are part of the IEEE-488 standards, define an instrument's ability to support various interface functions and should not be confused with programming commands found elsewhere in this manual. The interface function codes for the Model 2510 are listed in Table C-6. The codes define Model 2510 capabilities as follows:

Table C-6
Model 2510 interface function codes

Code	Interface function
SH1	Source Handshake capability
AH1	Acceptor Handshake capability
T5	Talker (basic talker, serial poll, unaddressed to talk on LAG)
L4	Listener (basic listener, unaddressed to listen on TAG)
SR1	Service Request capability
RL1	Remote/Local capability
PP0	No Parallel Poll capability
DC1	Device Clear capability
DT1	Device Trigger capability
C0	No Controller capability
E1	Open collector bus drivers
TE0	No Extended Talker capability
LE0	No Extended Listener capability

SH (Source Handshake Function) — SH1 defines the ability of the instrument to initiate the transfer of message/data over the data bus.

AH (Acceptor Handshake Function) — AH1 defines the ability of the instrument to guarantee proper reception of message/data transmitted over the data bus.

T (Talker Function) — The ability of the instrument to send data over the bus to other devices is provided by the T function. Instrument talker capabilities (T5) exist only after the instrument has been addressed to talk.

L (Listener Function) — The ability for the instrument to receive device-dependent data over the bus from other devices is provided by the L function. Listener capabilities (L4) of the instrument exist only after it has been addressed to listen.

SR (Service Request Function) — SR1 defines the ability of the instrument to request service from the controller.

RL (Remote-Local Function) — RL1 defines the ability of the instrument to be placed in the remote or local modes.

PP (Parallel Poll Function) — The instrument does not have parallel polling capabilities (PP0).

DC (Device Clear Function) — DC1 defines the ability of the instrument to be cleared (initialized).

DT (Device Trigger Function) — DT1 defines the ability of the Model 2510 to have readings triggered.

C (Controller Function) — The instrument does not have controller capabilities (C0).

TE (Extended Talker Function) — The instrument does not have extended talker capabilities (TE0).

LE (Extended Listener Function) — The instrument does not have extended listener capabilities (LE0).

E (Bus Driver Type) — The instrument has open-collector bus drivers (E1).

D
IEEE-488 and SCPI Conformance Information

Introduction

The IEEE-488.2 standard requires specific information about how the Model 2510 implements the standard. Paragraph 4.9 of the IEEE-488.2 standard (Std 488.2-1987) lists the documentation requirements. Table D-1 provides a summary of the requirements, and provides the information or references in the manual for that information. Table D-2 lists the coupled commands used by the Model 2510.

The Model 2510 complies with SCPI version 1996.0. Tables 10-1 through 10-10 in Section 10 list the SCPI confirmed commands and the non-SCPI commands implemented by the Model 2510.

Table D-1
IEEE-488 documentation requirements

	Requirements	Description or reference
(1)	IEEE-488 Interface Function Codes.	See Appendix C.
(2)	Behavior of Model 2510 when the address is set outside the range 0-30.	Cannot enter an invalid address.
(3)	Behavior of Model 2510 when valid address is entered.	Address changes and bus resets.
(4)	Power-On Setup Conditions.	Determine by :SYSTem:POSetup. See Section 3.
(5)	Message Exchange Options:	
(a)	Input buffer size.	2048 bytes.
(b)	Queries that return more than one response message unit.	None.
(c)	Queries that generate a response when parsed.	All queries (Common Commands and SCPI).
(d)	Queries that generate a response when read.	None.
(e)	Coupled commands.	See Table D-2.
(6)	Functional elements required for SCPI commands.	Contained in SCPI command subsystems tables (see Tables 10-1 through 10-10).
(7)	Buffer size limitations for block data.	Block display messages: 32 characters max.
(8)	Syntax restrictions.	See Section 6, "Programming syntax."
(9)	Response syntax for every query command.	See Section 6, "Programming syntax."
(10)	Device-to-device message transfer that does not follow rules of the standard.	None.
(11)	Block data response size.	See Section 10, "FORMat subsystem."
(12)	Common Commands implemented by Model 2510.	See Section 8, "Common commands."
(13)	Calibration query information.	See Service Manual.
(14)	Trigger macro for *DDT.	Not applicable.
(15)	Macro information.	Not applicable.
(16)	Response to *IDN (identification).	See Section 8, "Common commands."
(17)	Storage area for *PUD and *PUD?.	Not applicable.
(18)	Resource description for *RDT and *RDT?.	Not applicable.
(19)	Effects of *RST, *RCL and *SAV.	See Section 8, "Common commands."
(20)	*TST information.	See Section 8, "Common commands."
(21)	Status register structure.	See Section 7, "Status structure."
(22)	Sequential or overlapped commands.	All are sequential except :INIT.
(23)	Operation complete messages.	*OPC, *OPC? and *WAI; see Section 8, "Common commands."

Table D-2
Coupled commands

Command	Also changes
SENS:TEMP:RTD:TYPE (except for USER type)	SENS:TEMP:RTD:ALPHa SENS:TEMP:RTD:BETA SENS:TEMP:RTD:DELTa
SENS:RES:CURR or SENS:TEMP:CURR	SENS:RES:CURR:AUTO to OFF SENS:TEMP:CURR:AUTO to OFF
SOUR:CLE:IMM	OUTP:STAT to OFF
REN, GTL	See local and remote transition in Section 6, "Differences: remote vs. local operation."

E
Example Programs

Introduction

This appendix includes two example BASIC programs to demonstrate basic Model 2510 remote operation:

- Program 1: Basic temperature control.
- Program 2: Using setpoint tolerance to generate a temperature sweep.

Program requirements

Computer hardware requirements

The following computer hardware is required to run the example programs:

- IBM PC compatible computer.
- Keithley KPC-488.2, KPS-488.2, or KPC-488.2AT, or CEC PC-488 IEEE-488 interface for the computer.
- Shielded IEEE-488 interface cable (Keithley Model 7007).

Software requirements

In order to use the example programs, you will need the following computer software:

- Microsoft QBasic (supplied with MS-DOS 5.0 or later).
- MS-DOS version 5.0 or later or Windows 95/98.
- HP-style Universal Language Driver, CECHP.EXE (supplied with Keithley and CEC interface cards listed above).

General program instructions

1. With the power off, connect the Model 2510 to the IEEE-488 interface of the computer. Be sure to use a shielded IEEE-488 cable for bus connections.
2. Turn on the computer and the Model 2510.
3. Make sure the Model 2510 is set for a primary address of 15. (Use the front panel MENU/COMMUNICATIONS/GPIB selection to check or change the address.)
4. Make sure that the computer bus driver software (CECHP.EXE) is properly initialized.
5. Enter the QBasic editor, and type in the example program from the listing later in this Appendix.
6. Check thoroughly for errors, then save it using a convenient filename.
7. Run the program, and note any display messages on the computer screen.

Basic temperature control program

Program 1 is a complete program listing for the basic remote control-measure sequence outlined in Section 3.

Instrument setup

Briefly, Program 1 sets up the Model 2510 as follows:

- Temperature function, setpoint 50°C.
- Lower and upper limits: 10°C and 100°C
- PT100 RTD sensor with 4-wire sensing.
- PID loop constants: 10, 0.5, and 0.
- Current limit: 5.25A.

Requested readings

The program requests the following readings from the unit:

- TEC power.
- TEC current.
- Temperature.
- TEC DC resistance.
- TEC voltage.

Setpoint tolerance temperature sweep

Overview

As discussed in detail in Section 3, the setpoint tolerance feature allows the unit to generate an SRQ when the measured temperature is within a certain percent tolerance of the setpoint for a specific number of readings. This feature can be used to generate temperature sweeps, as we will now discuss.

Figure E-1 demonstrates the basic method for generating temperature sweeps. Once the temperature stabilizes at the Initial Setpoint, a step change is made in the setpoint setting. Once the temperature is within the programmed setpoint tolerance, the Model 2510 generates an SRQ to the computer to indicate that the new setpoint has been reached. The process is then repeated for as many steps in the temperature sweep as required.

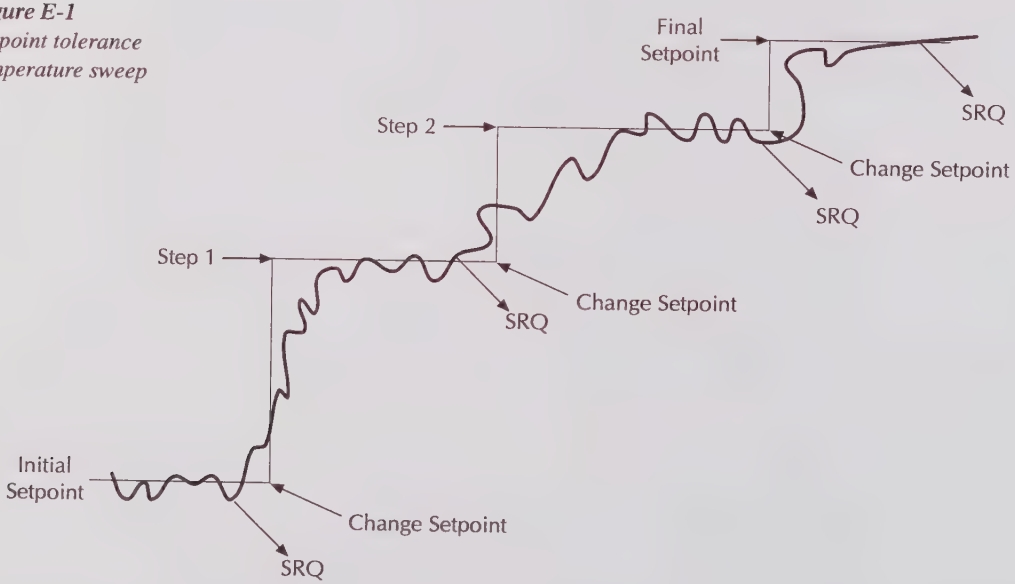


Figure E-1
Setpoint tolerance temperature sweep

Enabling SRQ on setpoint tolerance

To enable an SRQ when the setpoint tolerance has been reached, send the following commands:

- :STAT:MEAS:ENABLE 4096 – Enable bit 12, SPT (setpoint tolerance) in the Measurement Event Enable Register.
- *SRE 1 – Enable the MSB (Message Summary Bit) in the Service Request Enable Register.

See Section 7 for status register and SRQ details.

Temperature sweep program summary

Program 2 demonstrates the basic programming steps for generating a temperature sweep using setpoint tolerance. For the most part, the instrument is operated using default conditions by sending a *RST command at the beginning of the program. (See Section 1, Table 1-3.) Other key programming modes include:

- SRQ on setpoint tolerance enabled.
- Setpoint tolerance percent, window: 1%, 10 reading count.
- Upper temperature limit: 90°C.
- Temperature setpoint sweep under program control: 40°C to 80°C in 10°C steps.

When the program is run, it cycles through the temperature setpoints, stopping to wait for the Model 2510 to an SRQ when each setpoint is within tolerance. The TEC power at each setpoint is also requested and displayed.

Program 1: Basic temperature control

```
' The program performs basic temperature control-measure procedure.
' 2510 primary address = 15.
OPEN "IEEE" FOR OUTPUT AS #1                    ' Open IEEE-488 output path.
OPEN "IEEE" FOR INPUT AS #2                     ' Open IEEE-488 input path.
PRINT #1, "INTERM CRLF"                         ' Set input terminator.
PRINT #1, "OUTTERM LF"                          ' Set output terminator.
PRINT #1, "REMOTE 15"                           ' Put 2510 in remote.
'
' Step 1: Restore defaults.
PRINT #1, "OUTPUT 15;*RST"                      ' Restore GPIB defaults.
'
' Step 2: Select function.
PRINT #1, "OUTPUT 15;:SOUR:FUNC TEMP"           ' Select temperature function.
'
' Step 3: Configure selected function.
PRINT #1, "OUTPUT 15;:UNIT:TEMP CEL"            ' Celsius temperature units.
PRINT #1, "OUTPUT 15;:SOUR:TEMP:PROT 100"       ' Upper limit = 100 deg.
PRINT #1, "OUTPUT 15;:SOUR:TEMP:PROT:LOW 10"    ' Lower limit = 10 deg.
PRINT #1, "OUTPUT 15;:TEMP:TRAN RTD"            ' Select RTD sensor type.
PRINT #1, "OUTPUT 15;:TEMP:RTD:TYPE PT100"      ' Select PT100 RTD sensor.
PRINT #1, "OUTPUT 15;:TEMP:CURR:AUTO ON"        ' Use default RTD sensor current.
PRINT #1, "OUTPUT 15;:SYST:RSEN ON"             ' Enable 4-wire sensing.
PRINT #1, "OUTPUT 15;:SOUR:TEMP:LCON 10"        ' Gain (P) constant = 10.
PRINT #1, "OUTPUT 15;:SOUR:TEMP:LCON:INT 0.5"   ' Integral (I) constant = 0.5.
PRINT #1, "OUTPUT 15;:SOUR:TEMP:LCON:DER 0"     ' Derivative (D) constant = 0.
'
' Step 4: Program setpoint.
PRINT #1, "OUTPUT 15;:SOUR:TEMP 50"             ' Setpoint = 50 deg. C.
'
' Step 5: Program current limit.
PRINT #1, "OUTPUT 15;:SENS:CURR:PROT MAX"       ' Current limit = 5.25A.
'
' Step 6: Turn on output.
PRINT #1, "OUTPUT 15;:OUTP ON"
'
' Step 7: Request readings.
CLS
PRINT "Waiting 30 seconds for TEC to stabilize..."
SLEEP 30
PRINT #1, "OUTPUT 15;:MEAS:POW?"                ' Query TEC power.
PRINT #1, "ENTER 15"
INPUT #2, R
PRINT "TEC power:"; R
PRINT #1, "OUTPUT 15;:MEAS:CURR?"               ' Query TEC current.
PRINT #1, "ENTER 15"
INPUT #2, R
PRINT "TEC current:"; R
PRINT #1, "OUTPUT 15;:MEAS:TEMP?"               ' Query temperature.
PRINT #1, "ENTER 15"
INPUT #2, R
```

```
PRINT "Temperature:"; R
PRINT #1, "OUTPUT 15;:MEAS:RES?"              ' Query TEC DC resistance.
PRINT #1, "ENTER 15"
INPUT #2, R
PRINT "TEC DC resistance:"; R
PRINT #1, "OUTPUT 15;:MEAS:VOLT?"             ' Query TEC voltage.
PRINT #1, "ENTER 15"
INPUT #2, R
PRINT "TEC voltage:"; R
'
'Step 8: Turn off output.
PRINT #1, "OUTPUT 15;:OUTP OFF"
PRINT "Done."
END
```

Program 2: Setpoint tolerance temperature sweep

```
' The program uses setpoint tolerance to generate a temperature sweep.
' 2510 primary address = 15.
OPEN "IEEE" FOR OUTPUT AS #1                        ' Open IEEE-488 output path.
OPEN "IEEE" FOR INPUT AS #2                         ' Open IEEE-488 input path.
PRINT #1, "INTERM CRLF"                             ' Set input terminator.
PRINT #1, "OUTTERM LF"                              ' Set output terminator.
PRINT #1, "REMOTE 15"                               ' Put 2510 in remote.
PRINT #1, "OUTPUT 15;*RST"                          ' Restore GPIB defaults.
PRINT #1, "OUTPUT 15;:STAT:MEAS:ENAB 4096;*SRE 1" ' Enable SRQ on SPT.
PRINT #1, "OUTPUT 15;:SOUR:STOL 1"                  ' 1% setpoint tolerance.
PRINT #1, "OUTPUT 15;:SOUR:STOL:COUN 10"            ' 10 reading tolerance window.
PRINT #1, "OUTPUT 15;:SOUR:TEMP:PROT 90"            ' Upper temp. limit = 90 deg.
PRINT #1, "OUTPUT 15;:OUTP ON"                      ' Turn on output.
CLS
FOR Setpoint = 40 TO 80 STEP 10                     ' Loop for 40 to 80 deg. setpoints.
PRINT #1, "OUTPUT 15;:SOUR:TEMP" ; Setpoint         ' Program temp. setpoint.
PRINT "Waiting for"; Setpoint; "degree setpoint to stabilize..."
DO: PRINT #1, "SRQ?"                                ' Request SRQ status.
        INPUT #2, S                                 ' Input SRQ status byte.
LOOP UNTIL S                                        ' Wait for setpoint tolerance.
PRINT #1, "OUTPUT 15;:*CLS"                         ' Clear SPT.
PRINT #1, "SPOLL 15"                                ' Serial poll unit to clear SRQ.
INPUT #2, S
PRINT #1, "OUTPUT 15;:MEAS:POW?"                    ' Query TEC power reading.
PRINT #1, "ENTER 15"
INPUT #2, Power
PRINT "TEC power at"; Setpoint; "degree setpoint:"; Power
NEXT Setpoint
PRINT #1, "OUTPUT 15;:OUTP OFF"                     ' Turn off output.
PRINT "Done."
END
```

F
GPIB 488.1 Protocol

Introduction

The SourceMeter supports two GPIB protocols: SCPI and 488.1. The 488.1 protocol is included to significantly increase speed over the GPIB.

When using the 488.1 protocol, throughput is enhanced up to 10 times for data sent to the SourceMeter (command messages) and up to 20 times for data returned by the SourceMeter (response messages). The speed of readings sent over the GPIB is also increased.

NOTE *With the 488.1 protocol selected, you will still use SCPI commands to program the SourceMeter. Operation differences between the two protocols are discussed in this appendix.*

Selecting the 488.1 protocol

Perform the following steps to select the 488.1 protocol:

1. Press MENU to display the MAIN MENU.
2. Place the cursor on COMMUNICATION, and press ENTER to display the COMMUNICATIONS SETUP menu.
3. Place the cursor on GPIB, and press ENTER to display the present GPIB address.
4. Press ENTER to display the GPIB PROTOCOL menu.
5. Place the cursor on 488.1, and press ENTER.
6. Use the EXIT key to back out of the menu structure.

When switching between the SCPI protocol and 488.1 protocol, the instrument does not reset. The GPIB protocol setting is saved in EEPROM, and the unit will power up with that selected protocol.

The GPIB protocol cannot be changed over the bus. However, there is a query command to determine the presently selected protocol. When the 488.1 protocol is selected, the message exchange protocol (MEP) disables. Therefore, if you use the following query to request the state of MEP, you will know which protocol is enabled:

:SYSTem:MEP[:STATe]?

If a "1" is returned, MEP is enabled, and the SCPI protocol is selected. A "0" indicates that MEP is disabled, and the 488.1 protocol is enabled. To summarize:

1 = SCPI protocol
0 = 488.1 protocol

Protocol differences

The following information covers the differences between the 488.1 protocol and the SCPI protocol.

Message exchange protocol (MEP)

When the 488.1 protocol is selected, the MEP is disabled to speed up GPIB operation.

The following guidelines/limitations must be followed when using the 488.1 protocol:

- If a query is sent, it must be the only command on the line (this limitation also means no multiple queries can be sent). Otherwise, full SCPI command syntax is still supported including long-form and short form commands, multiple commands, and MIN/MAX/DEF parameter definitions.

  For example, the following command strings are invalid:
  :SOUR:VOLT 10;*OPC?
  :SOUR:RES?;:READ?
  :READ?;:READ?

  The following command strings are valid:
  :SOURce1:TEMPerature 25;:sour:temp:prot 50
  :sens:curr:prot 2;:sour:volt 5
  :SENS:TEMP:CURR? MAX
  :READ?
- When a query is sent, either the data must be read back or a Device Clear (DCL) or Interface Clear (IFC) must be performed to reset the query.
- When sending a command or query, do not attempt to read data from the SourceMeter until the terminator has been sent (usually Line Feed with EOI). Otherwise, a DCL or IFC must be sent to reset the input parser.
- When receiving data, all data, up to and including the terminator (LF with EOI), must be accepted. Otherwise, a DCL or IFC must be sent to reset the output task.
- Empty command strings (terminator only) should not be sent.

Using SCPI-based programs

In general, an existing SCPI-based program will run properly and faster in the 488.1 protocol as long as it meets the above guidelines and limitations.

Bus hold-off

*OPC, *OPC?, and *WAI are still functional but are not needed for the 488.1 protocol. When sending commands, the GPIB is automatically held off when it detects a terminator. The hold-off is released when all the commands have finished executing, or if there is some parser or command error. An exception is an initiate command, which releases the hold-off immediately and does not wait for all of the readings to be acquired. This immediate release of bus hold-off is done to support GET, SDC, IFC, *TRG, *RCL, *RST, SYSTem:PRESet and ABORt during data acquisition.

Trigger-on-talk

Trigger-on-talk functionality has been added for the 488.1 protocol. If a query has not been received by the instrument, the SourceMeter will automatically assume a READ? command has been sent when it is addressed to talk. This technique increases GPIB speed by decreasing the transmission and parser times for the command. Remember that the output must be on (:OUTput:STATe ON) before you can take readings.

Message available

The MAV (message available) bit in the Serial Poll byte will be set when the query is finished being processed, not when there is data available in the output buffer (as with the SCPI protocol). For the 488.1 protocol, output data will not be formatted until the first request for data is received. This delay may cause unexpected time-outs when using SRQ on MAV for queries that take a long time to execute.

General operation notes

- The TALK, LSTN, and SRQ annunciators are not functional in the 488.1 protocol. This speeds up data throughput greatly. The REM annunciator still operates since it is critical to fundamental GPIB operation.
- If the unit is in REMote, the GTL command may not put the SourceMeter into the local mode. Only the front panel LOCAL key is guaranteed to operate, if not in local lockout (LLO). GTL will still disable LLO.
- IEEE-488 bus commands and features (GET, IFC, SDC, DCL, LLO, SPE, SPD (Serial Poll), and SRQ) are still fully supported.
- Multiple TALKs on the same query are supported as in the SCPI protocol. This feature is useful when reading back long ASCII strings.

Index

Symbols

:DISPlay subsystem commands 10-3
:FORMat subsystem commands 10-4
*IDN? — identification query 8-3
*OPC — operation complete 8-3
*OPC? — operation complete query 8-3
*OPT? — option query 8-3
*RCL — recall 8-4
*RST — reset 8-4
*SAV — save 8-4
:SOURce 10-7
:SOURce2 subsystem commands 10-11
:STATus subsystem commands 10-11
:SYSTem subsystem commands 10-12
*TRG — trigger 8-4
*TST? — self-test query 8-4
:UNIT subsystem commands 10-13
*WAI — wait-to-continue 8-5

Numerics

16-bit status register 7-5
2-wire connections 2-2
4-wire connections 2-3
+5V output 5-3

A

AC resistance 3-2, 9-3
Acquiring readings 9-2
Acquiring readings
    :FETCh? 9-2
    :INITiate 9-4
    :MEASure 9-3
    :READ? 9-4
Address commands C-8
Addressed multiline commands C-8
Adjusting setpoints 3-9
ASCII data 10-18

B

Basic Operation 3-1
Basic temperature control program E-3
Bus commands C-6
Bus hold-off F-4
Bus lines C-4
Bus management lines C-4

C

Cables and adapters 1-3
Carrying case 1-4
Case sensitivity 6-14
Clearing registers and queues 7-4
Command codes C-9, C-10
Command execution rules 6-17
Command path rules 6-17
Command reference 8-3
Command words 6-12
Commands and command parameters 6-13
Common Commands 8-1
    Command summary 8-2
Common commands C-9
Condition registers 7-17
Configuring
    current 3-5
    functions 3-3
    output 3-8
    resistance 3-6
    setpoint tolerance 3-8
    temperature 3-3
    voltage 3-5
Connections 2-1
Contact information 1-2
Control characteristics 3-2
Controlling digital output lines 5-4
Coupled commands D-4
Current source configuration menu 1-25, 3-5

D

Data format commands for reading status registers 7-6
Data lines C-4
DC resistance 9-3
DCL (device clear) 6-12
Default settings 1-17
Digital I/O port 5-2
Digital I/O Port and Output Enable 5-1
Digital output configuration 5-3
Digital output lines 5-3
Display 1-11
    examples 1-14
    format 1-11
    Reading format 1-13

readings 1-12
Current function 1-12
Resistance function 1-13
Temperature function 1-12
Voltage function 1-12
DISPlay subsystem 10-14
ASCII display values 10-16
Control display 10-14
Define :TEXT messages 10-15
Read display 10-14
DISPLAY TOGGLE/LOCAL Key 1-6, 1-11, 6-10

E

EDIT keys 1-16
Eliminating common SCPI errors B-7
-113, "Undefined header" B-7
-410, "Query INTERRUPTED" B-7
-420, "Query UNTERMINATED" B-8
Enabling SRQ on setpoint tolerance E-5
Error queue 7-20
Error queue commands 7-21
Event enable registers 7-18
Event enable registers commands 7-18
Event registers 7-17
Example Programs E-1

F

Factory default settings 1-17
FORMat subsystem 10-18
Byte order 10-22
Data elements 10-20
Data format 10-18
Status register format 10-23
Front and rear panel familiarization 1-6
Front panel
Basic control-measure procedure 3-10
Basic remote control-measure procedure 3-11
GPIB operation 6-9
DISPLAY TOGGLE/LOCAL key 6-10
Error and status messages 6-9
GPIB status indicators 6-9
summary 1-6
Front panel tests 1-16
Fuse
replacement 1-11

G

General bus commands 6-11
General information 1-2
General operation notes F-4
GET (group execute trigger) 6-12
Getting Started 1-1
GPIB
connections 6-4
operation 6-3
standards 6-3
GPIB 488.1 Protocol F-1
GPIB status indicators
LSTN 6-10
REM 6-9
SRQ 6-10
TALK 6-10
Ground connect mode 2-6
GTL (go to local) 6-12

H

Handshake lines C-5

I

IEEE command groups C-12
IEEE-488 and RS-232 connector locations 6-5
IEEE-488 and SCPI Conformance
Information D-1
Introduction D-2
IEEE-488 bus command summary C-6
IEEE-488 bus configuration C-3
IEEE-488 Bus Overview C-1
Bus description C-2
Introduction C-2
IEEE-488 connections 6-4
IEEE-488 connector 6-4
IEEE-488 documentation requirements D-3
IEEE-488 handshake sequence C-5
IEEE-488.2 common commands and queries 8-2
IFC (interface clear) 6-11
INPUT sensing 2-4
Input/output connections 2-2
Input/output connector 2-2
Inspection 1-3
Instrument setup E-3
Interface function codes C-13
Introduction E-2, F-2

L

Leading colon 6-15
Limit characteristics 3-3
Line frequency setting 1-10
Line power connection 1-9
LLO (local lockout) 6-11
Local-to-remote transition 6-2
Long-form and short-form versions 6-15

M

Main menu tree 1-23
Manual addenda 1-2
Measurement characteristics 3-2
Measurement event register 7-14
Measurement event status 7-15
Menus 1-20
  Configuration menus 1-22
  Main menu 1-20
  Rules to navigate menus 1-22
Message available F-4
Message exchange protocol 6-18
Message exchange protocol (MEP) F-3
Model 2510 front panel 1-6
  Annunciators 1-7
  EDIT keys 1-6
  Function keys 1-6
  Handle 1-7
  Operation keys 1-6
  Power controls 1-7
Model 2510 interface function codes C-13
Model 2510 rear panel 1-8
  Digital I/O port 1-8
  GPIB connector 1-8
  Ground screw 1-8
  Input/output connector 1-8
  Power module 1-8
  RS-232 connector 1-8
  Trigger link connector 1-8
Model 2510 status register structure 7-3
Multiple command messages 6-16
Multiple response messages 6-18

O

ON/OFF indicator 1-16
Operation event register 7-13
Operation event status 7-13
Operation overview 3-2
Options and accessories 1-3
Output configuration menu 1-26, 3-8
Output enable line 5-3, 5-6
  Activating output enable 5-6
Output queue 7-19
OUTPUT sensing 2-4
OUTPut subsystem 10-23
  Output enable line control 10-24
  Turn source on or off 10-23
OUTPut subsystem commands 10-4
Overflow reading 10-20

P

PC serial port pinout 6-9
PID control chara cteristics 4-5
PID Control Concepts 4-1
PID tuning 4-5
Port configuration 5-2
Power-on configuration 1-17
Power-up 1-9
Power-up sequence 1-9
Primary address 6-5
Product overview 1-5
Program 1
  Basic temperature control E-6
Program 2
  Setpoint tolerance temperature sweep E-7
Program message terminator (PMT) 6-17
Program messages 6-16
Program requirements E-2
  Computer hardware E-2
  General program instructions E-2
  Software E-2
Programming and reading registers 7-5
Programming enable registers 7-5
Programming example
  program and read register set 7-19
  read error queue 7-21
  set MSS (B6) when error occurs 7-10
Programming syntax 6-12
Protection limits 3-13
  Current 3-14
  Resistance 3-14
  Temperature 3-14
  Voltage 3-14
Protocol differences F-3

Q

Query commands 6-14
Questionable event register 7-16
Questionable event status 7-16
Queues 7-2, 7-19

R

Rack mount kits 1-4
Reading registers 7-6
Rear panel summary 1-8
Reference tables 10-2
Register bit descriptions 7-11
Remote Operations 6-1
Remote vs. local operation differences 6-2
Remote-to-local transition 6-2
REN (remote enable) 6-11
Requested readings E-3
Resistance configuration menu 1-25, 3-7
Resistance measurement
    AC 3-2, 9-3
    DC 3-2, 9-3
Response message terminator (RMT) 6-18
Response messages 6-18
Restoring setups 1-17
RS-232 connector pinout 6-8
RS-232 interface connector 6-8
RS-232 interface operation 6-6
    Baud rate 6-6
    Data bits and parity 6-6
    Flow control (signal handshaking) 6-7
    RS-232 connections 6-8
    Sending and receiving data 6-6
    Terminator 6-7
RTD coefficients 4-6

S

Safety precautions 3-2
Safety symbols and terms 1-2
Saving and restoring user setups 1-17
Saving setups 1-17
SCPI Command Reference 10-1
SCPI commands C-9
SCPI Signal-Oriented Measurement
    Commands 9-1
    Command examples 9-4
    Command summary 9-2
SDC (selective device clear) 6-12
Selecting an interface 6-2
Selecting the 488.1 protocol F-2
Sending a response message 6-18
Sense selection 2-4
SENSe1 subsystem 10-24
    Current function 10-24
    Resistance function 10-25
    RTD sensor parameters 10-27
    Solid-state sensor parameters 10-30
    Temperature function 10-26
    Thermistor sensor parameters 10-29
Sensing considerations 2-4
    2-wire sensing 2-4
    4-wire sensing 2-5
Sensor coefficients 4-6
Serial polling and SRQ 7-9
Service request enable register 7-9
Setpoint adjustment ranges 3-9
Setpoint tolerance 3-15, 10-39
    indicator 3-16
    operation 3-16, 3-17
    Ranges 3-16
Setpoint tolerance temperature sweep E-4
Setting interface parameters 6-2
Short-form rules 6-15
Signal-oriented measurement command
    summary 9-2
Single command messages 6-16
Sink operation 5-3
SOURce 10-31, 10-32, 10-33, 10-35, 10-38
Source operation 5-4
SOURce2 subsystem 10-40
SPE, SPD (serial polling) 6-12, 7-9
Specifications A-1
Standard event register 7-11
Standard event status 7-12
Status and Error Messages B-1, B-2
    Introduction B-2
Status and error messages 1-16, B-3
Status byte and service request (SRQ) 7-7
Status byte and service request commands 7-10
Status byte and service request enable register
    commands 7-10
Status byte and SRQ 7-2
Status byte register 7-8
Status register sets 7-2, 7-11
Status Structure 7-1
    Overview 7-2
STATus subsystem 10-41
    Error queue 10-43
    Program event enable registers 10-42
    Read condition registers 10-42
    Read event registers 10-41
    Select default conditions 10-42
subsystem 10-31
    Control source output-off 10-31
    Current function 10-32
    Resistance function 10-33
    Select source function 10-31

Temperature function 10-35
Voltage function 10-38
subsystem commands 10-7
System identification 1-10
SYSTem subsystem 10-44
2-wire/4-wire sense mode 10-47
Default conditions 10-44
Error queue 10-45
Ground connect mode 10-48
Read version of SCPI standard 10-47
Reset timestamp 10-47
RS-232 interface 10-48
Select power line frequency setting 10-44
Simulate key presses 10-46

T

Temperature
control methods 4-2
On-off control 4-3
Proportional (P) control 4-3
Proportional-derivative (PD) control 4-4
Proportional-integral-derivative (PID) control 4-4
control model 4-2
Temperature configuration menu 1-24, 3-4
Temperature sensors 3-18
Open and shorted lead status indications 3-18
Solid-state sensor restrictions 3-18
Temperature sweep program summary E-5
Thermistor coefficients 4-6
Trigger subsystem 10-49
Abort source/measure cycle 10-49
Initiate control/measure cycle 10-49
Trigger subsystem commands 10-13
Trigger-on-talk F-4
Typical addressed common command sequence C-11
Typical addressed multiline command sequence C-11
Typical command sequences C-11

U

Unaddress commands C-8
Uniline commands C-7
UNIT subsystem 10-49
Universal multiline commands C-7
Using common and SCPI commands in the same message 6-17
Using output enable 5-7
Using SCPI-based programs F-3

V

Voltage source configuration menu 1-24, 3-5

W

Warranty information 1-2

KEITHLEY

Service Form

Model No. _____ **Serial No.** _____ **Date** _____

Name and Telephone No. _____

Company _____

List all control settings, describe problem and check boxes that apply to problem. _____

_____

_____

❑ Intermittent
❑ Analog output follows display
❑ Particular range or function bad; specify _____

❑ IEEE failure
❑ Obvious problem on power-up
❑ Batteries and fuses are OK

❑ Front panel operational
❑ All ranges or functions are bad
❑ Checked all cables

Display or output (check one)

❑ Drifts
❑ Unable to zero
❑ Unstable

❑ Overload
❑ Will not read applied input

❑ Calibration only
❑ Certificate of calibration required
❑ Data required

(attach any additional sheets as necessary)

Show a block diagram of your measurement including all instruments connected (whether power is turned on or not). Also, describe signal source.

Where is the measurement being performed? (factory, controlled laboratory, out-of-doors, etc.)_____

_____

What power line voltage is used? _____ Ambient temperature? _____ °F

Relative humidity? _____ Other? _____

Any additional information. (If special modifications have been made by the user, please describe.)

_____

_____

Be sure to include your name and phone number on this service form.

KEITHLEY

Keithley Instruments, Inc.
28775 Aurora Road
Cleveland, Ohio 44139
www.keithley.com

Printed in the U.S.A.

MP MASTER

MA28533